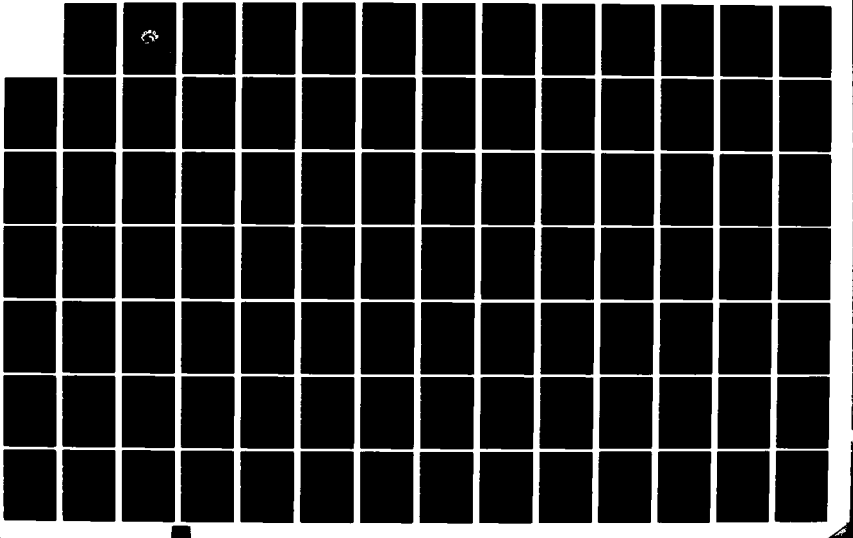
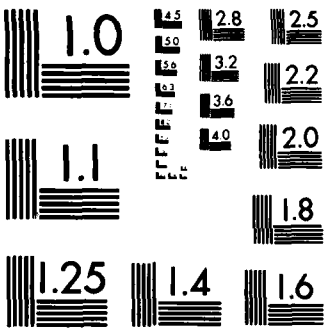
AD-A127 566 OPTICAL CHARACTER RECOGNITION FOR AUTOMATED CARTOGRAPHY: THE ADVANCED DEV..(U) NAVAL OCEAN RESEARCH AND DEVELOPMENT ACTIVITY NSTL STATION MS..

1/2

UNCLASSIFIED R M BROWN ET AL. MAR 83 NORDA-TN-187 F/G 8/2 NL





(12)

NORDA Technical Note 187

Naval Ocean Research and Development Activity
NSTL Station, Mississippi 39529

Optical Character Recognition for Automated Cartography: The Advanced Development Handprinted Symbol Recognition System

Prepared for:
NORDA Code 550, Mapping, Charting and Geodesy Program Management Office

Sponsored by:
Defense Mapping Agency HQ STT

R.M. Brown
Mapping, Charting and Geodesy Division
Ocean Science and Technology Laboratory

C.F. Cheng
Computer Sciences Corporation

March 1983

ABSTRACT

This Naval Ocean Research and Development Activity (NORDA) Technical Note reviews the recent progress and present status of the Defense Mapping Agency (DMA) Subtask "Optical Character Recognition Algorithm Development" being carried out in the NORDA Pattern Analysis Laboratory (PAL), Mapping, Charting, and Geodesy (MC&G) Division. In particular, it describes the Handprinted Symbol Recognition System that is capable of reading and digitizing a wide range of isolated, unconstrained (free-form handprinted numerals appearing on various DMA manuscript documents; e.g., smooth sheets, digital feature analysis data (DFAD) reference sheets, interval numbers on contour elevation sheets, etc.). Finally, the fundamental shape measurement and recognition tools incorporated in the HSR System can provide the foundation for other DMA systems to read the alphabet, foreign diacritics, and other map and chart symbols.

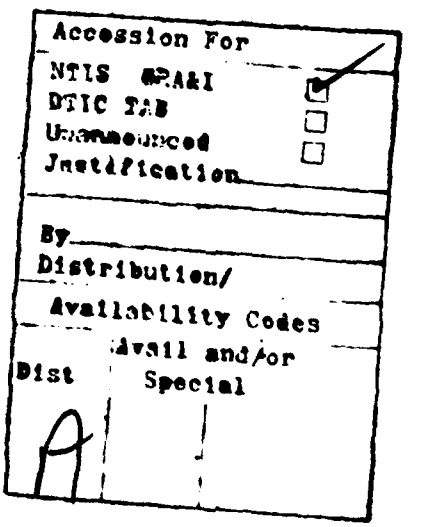
This NORDA Technical Note is composed of five chapters. The first chapter presents an overview of optical character recognition (OCR) and its relation to the automated cartography environment. It provides the DMA Subtask objectives and discusses them in the light of "symbol digitizing" and information transformations. The division of a "total OCR system" into data acquisition/document management and isolated character recognition is considered along with NORDA's recent tasking (FY-82) to integrate these two parts of the system to form a prototype for DMA production centers. Chapter Two presents a discussion of the different ways in which recognition systems are constructed. In particular, it considers the differences in approach necessary for constrained and free-form OCR. Chapter Three describes the DMA environment in which a handprinted OCR system must operate and discusses performance requirements. The general structure of the Handprinted Symbol Recognition System is outlined in Chapter Four. This material considers the key issues of information content, problems in the thinning or "vectorization" of a character, shape measurement and feature extraction, and finally character recognition or labeling. The interaction between each of these elements is emphasized. Chapter Five provides a brief summary of the current Subtask accomplishments and status along with areas where work is in progress toward developing other handprinted OCR capabilities for DMA.

DMA has recently indicated that "the whole area of artificial intelligence, pattern recognition and image processing and its application to hydrographic charting needs to be aggressively pursued" (Martin, 1982). The

advanced development efforts reported here directly address this DMA need and have made significant progress, over the past several years, on the very difficult problem of recognizing unconstrained handprinting on hydrographic smooth sheets and other graphic manuscripts containing numbers.

ACKNOWLEDGMENTS

This work was sponsored by DMA under Program element 63701B, with the Subtask title "Optical Character Recognition Algorithm Development," and was performed by NORDA Code 550, the Mapping, Charting and Geodesy Program Management Office. The DMA Program manager was Mr. Robert Penney, and LCDR Vic Hultstrand (Code 550) was the Project Manager. Dr. Robert M. Brown (Code 371) was the principal investigator, and Dr. Charles L. Walker (Code 371) was the co-investigator. Mr. C. F. Cheng of Computer Sciences Corporation was under contract to NORDA Code 370 for a significant portion of the work reported here.



CONTENTS

ILLUSTRATIONS

TABLES

OPTICAL CHARACTER RECOGNITION FOR AUTOMATED CARTOGRAPHY: THE ADVANCED DEVELOPMENT HANDPRINTED SYMBOL RECOGNITION SYSTEM

1.0 OCR ALGORITHM DEVELOPMENT AND THE AUTOMATED CARTOGRAPHIC PROCESS:AN OVERVIEW

1.1 DMA SUBTASK OBJECTIVE

This Naval Ocean Research and Development Activity (NORDA) Technical Note reviews the recent progress and present status of the Defense Mapping Agency (DMA) Subtask "Optical Character Recognition Algorithm Development" being carried out in the NORDA Pattern Analysis Laboratory (PAL), Mapping, Charting, and Geodesy (MC&G) Division. In particular, it describes the software package called the Handprinted Symbol Recognition (HSR) System. HSR, Version 1.0, was delivered to DMA in October 1980; an upgraded, advanced version of the character recognizer, HSR, Version 2.0, was completed in August 1982. This system is capable of reading and digitizing a wide range of isolated, unconstrained (free-form), handprinted numerals appearing on various DMA manuscripts, documents; e.g., smooth sheets, digital feature analysis data (DFAD) reference sheets, interval numbers on contour elevation sheets, etc. Finally, the fundamental shape measurement and recognition tools incorporated in the HSR System provide the foundation for other DMA systems to read the alphabet, foreign diacritics, and other map and chart symbols.

The character recognition system, HSR, Version 2.0, has been optimized to take advantage of the advanced character image preprocessing algorithms developed as part of NORDA's new DMA task assignment (FY-82) to integrate technologies from the Optical Character Recognition (OCR) and the Raster Scan Character Recognition (RSCR) development efforts (Walker, et al., 1983). These preprocessing algorithms will be presented in the NORDA Technical Note 210, "Image Preprocessing of Handprinted Symbols for Automated Cartography" (Brown, in prep.).

The objective of the OCR Algorithm Development Subtask "is to provide DMA with the capability to digitize and identify by computer automated techniques a wide variety of symbols involved in DMA map and chart production. These handprinted characters consist of free-form, unconstrained alphanumerics and navigation symbols. The techniques and software developed under this Subtask will provide a means of converting these graphic (analog) items to digital (computer compatible) form and will be usable for isolated symbols on both chart manuscripts and other documents containing handprinted data. This development and application of pattern analysis technology to the recognition of handprinted symbols will improve DMA's capability to process cartographic, hydrographic, oceanographic, and navigational data and products. It will also provide an important link in the generation and database storage of automated cartographic information" (OCR, 1982).

The following Sections expand on the relationship of this Subtask to the overall automated cartographic map and chart reading problem.

1.2 THE GRAPHICS IMAGE DIGITIZING PROCESS

DMA maps and charts are complex, two-dimensional, visual graphic representations of militarily significant information about the environment. The information contained on these products provides an abstracted structure, or model, of the real world from which the data was collected: aerial and satellite imagery, many different kinds of surveys (hydrographic, topographic, magnetic, gravity, etc.), census data, books and records, etc. Until recently, DMA has stored the major part of the data required to revise such maps and charts or to produce new map and chart (M/C) products in the analog (graphic) form of earlier M/Cs, overlays, satellite and photographic imagery, and various other "paper records."

The continued use of M/C graphic (analog) media as the basic storage mechanism makes it difficult to automate the

(1) comparison of information in different "records," i.e., maps and charts,
(2) retrieval of information based on "key words" or particular topics, e.g., all cities that meet a certain criteria,
(3) updating of "M/C records" or the generation of new map and chart products,
(4) comparison of "M/C records" with new incoming data, especially satellite imagery.

The difficulty, of course, arises primarily because of the graphics/image form of the data/information storage versus an abstracted information/knowledge database format accessible by computer.

In order to automate its M/C production process, DMA is building new storage mechanisms (e.g., database schemes) based on all-digital information handling technology available through computer automated cartography (AC). A major problem or task in the transition to this AC environment is the extraction and transformation of the information contained in M/C analog records.

Traditionally, this data extraction and transformation process has, to a great extent, been performed by humans; e.g., through the use of overlays, pull-ups, photographic rescaling, manual compilation, feature analysis tagging, operator data selection and digitizing, etc. These manual procedures are very time-consuming and labor intensive. A schematic of this process is shown in Figure 1. One should note that the information feedback loop AA' in Figure 1 does not provide an appropriate storage mechanism for future automated information handling and management; in particular, it perpetuates the existing analog-to-digital conversion problem.

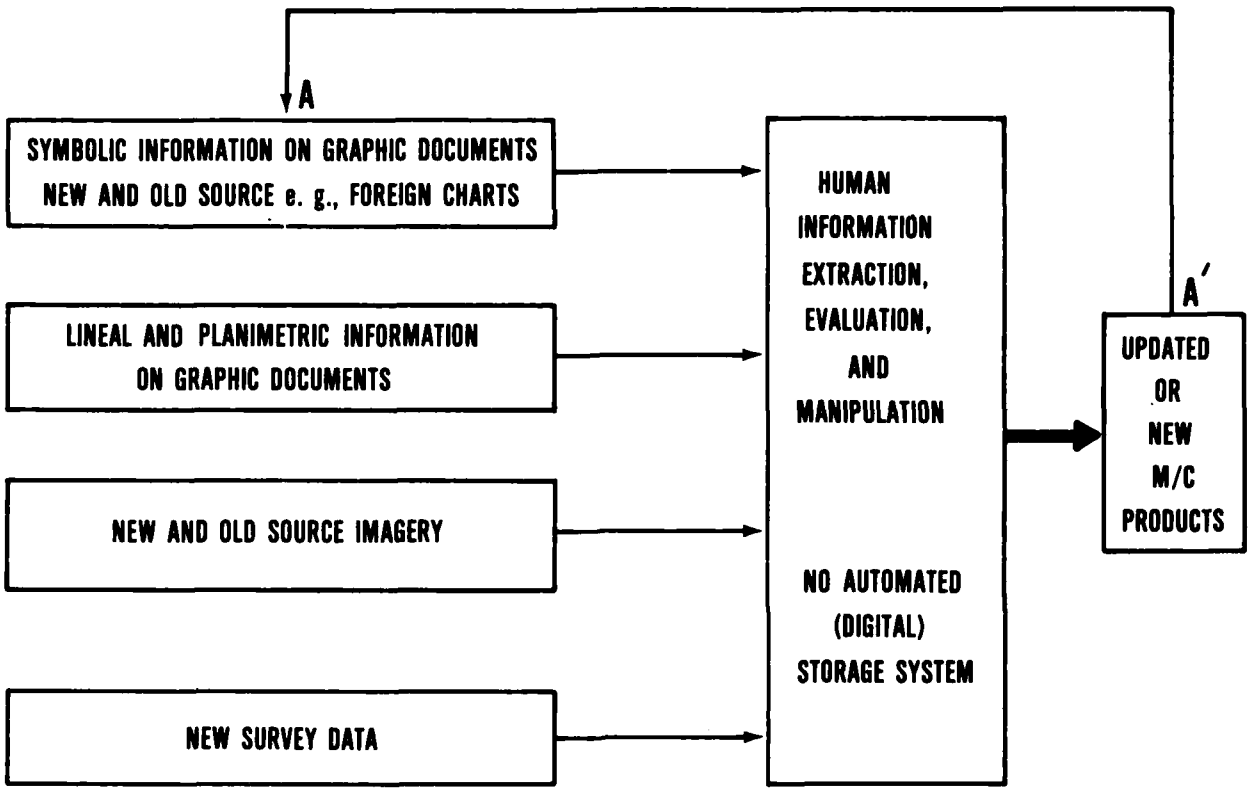


Figure 1. Schematic of manual information handling

Over the past several years, however, DMA has expended considerable efforts to automate various aspects of these processes and has the goal of becoming "all digital by 1990" (DMA 1982a; DMA 1982b). When completed, these efforts will lead to an AC processing concept like the one shown in general terms in Figure 2. Several blocks in this figure have already undergone considerable development; for example, raster scan technology is being employed to capture linear features on M/C products, block 2 (AGDS, ACDDS); techniques for automatic information extraction from imagery, block 3 (automatic scene segmentation/description, Digital Stereo Comparator/Compiler, automated feature tagging, Digital Pilot Operations); hydrographic information handling systems, block 4 (HIHANS); database technology and information/knowledge representation, storage, and retrieval, block 5 (Digital Pilot Operations).

1.3 AUTOMATED CARTOGRAPHY TRANSFORMATIONS: IMAGE-TO-INFORMATION-TO-IMAGE

The basic process just described is an information representation transformation process; i.e., the transformation of information from (1) the image/graphics database of M/C analog source documents to (2) the knowledge base of digital storage, retrieval, and manipulation and then back to (3) an image-base, visual presentation of new or updated products (softcopy or hardcopy).

A critical link or key to the automated cartographic information extraction and transformation task is the ability automatically to recognize symbols on the M/C analog documents and convert them and their associated attributes (including implicit information such as position, size, fonts, etc.) to the all-digital information storage schemes; this task is represented in Figure 2 by block 1. The overall technology required for this critical link can be divided into two parts:

(1) The global graphic/image data acquisition and document management task which includes

- raster scanning
- positional information capture
- generation of context (location/orientation) information
- character isolation
- "word" composition
- cartographic feature reference identification
- document description headers
- output formating for the targeted digital database;

(2) The task of recognizing isolated, free-form (unconstrained) characters which includes

- character preprocessing or "vectorization"
- "vectorization" or stroke generation
- use of context information

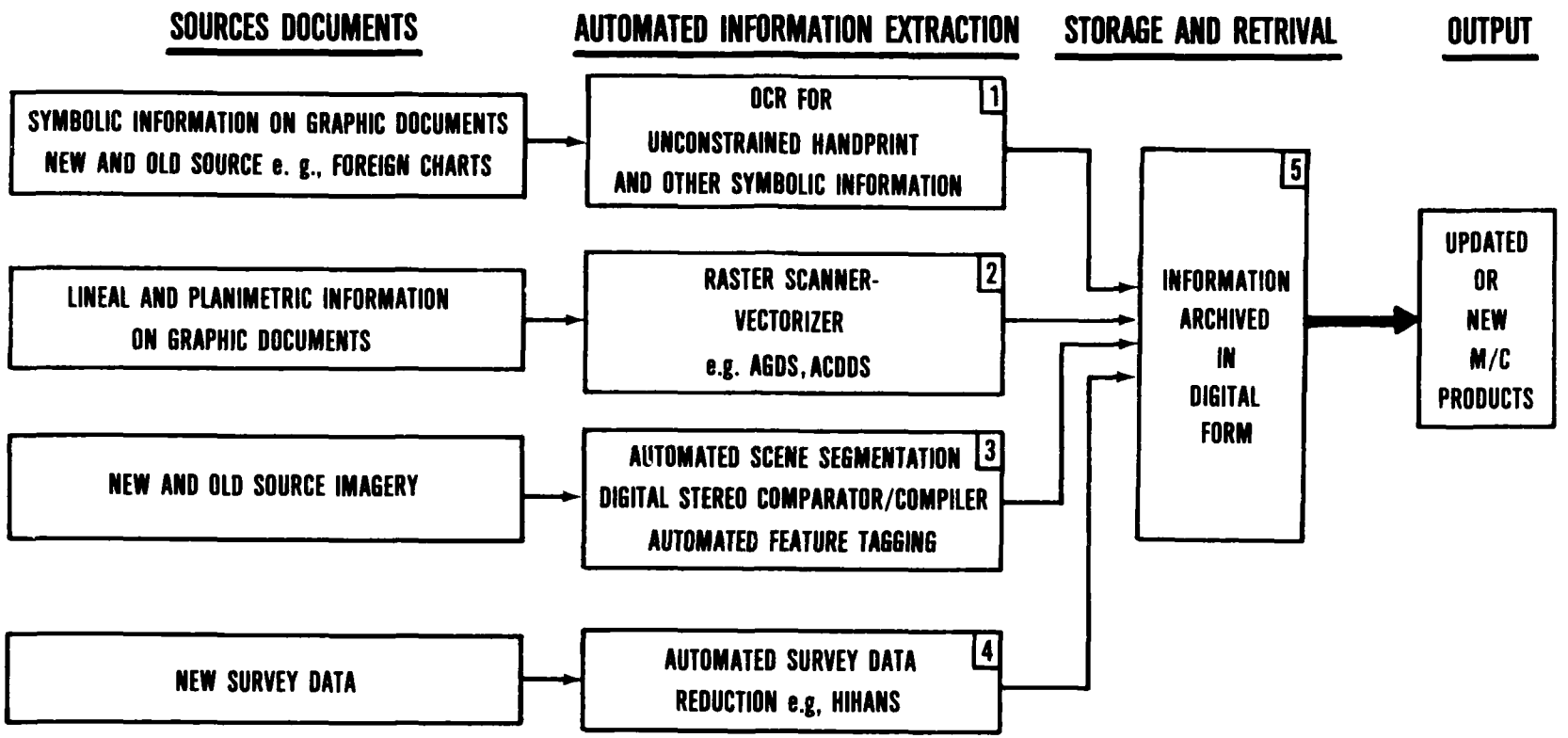


Figure 2. Schematic of automated cartography information extraction and storage

- shape measurement
- feature extraction
- decision/classification logic
- quality assurance

Division of the overall transformation process of character image-to-digital information codes into these two Subtasks was originally requested by DMA in FY-78 (OCR, 1978; RADC, 1978). Furthermore, the technologies which must be developed to solve these two parts of the transformation problem are quite different (at the initial level). Finally, each task in its own right is an appreciably sized development effort of considerable difficulty.

In FY-79, the NORDA Pattern Analysis Laboratory (PAL), Mapping, Charting, and Geodesy Division* initiated a comprehensive advanced development effort to solve the isolated, free-form (unconstrained) M/C handprinted character recognition problem under the DMA Subtask "Optical Character Recognition Algorithm Development." This effort was to build on the earlier NAVOCEANO/NORDA feasibility demonstrations in OCR (Bolton, 1977; Gronmeyer and Ruffin, 1978; Lybanon and Gronmeyer, 1978). At this same time (FY-79), DMA requested that the Rome Air Development Center (RADC) address the data acquisition and document management problem under the Subtask "Raster Scan Character Recognition." This RADC work was to be based on their earlier investigations into similar image/document handling technology.

1.4 NORDA ACCOMPLISHMENTS

The NORDA R&D activities have accomplished three significant goals since FY-79; they have

(1) successfully completed the development of a software package for recognizing isolated handprinted numerals; namely, the Handprinted Symbol Recognition (HSR) System, Version 2.0,
(2) laid specific technical ground work (testbed and algorithms) for free-form symbol recognition developments for a wide range of M/C analog documents, and
(3) established within the government a fundamental understanding of the unconstrained symbol recognition problem and a unique capability to develop (non-proprietary) software for this problem which can be expanded/extended to a wide variety of M/C symbolic information capture

* In 1979, the Division, NORDA Code 370, was called the MC&G Development Group, NORDA Code 302. The PAL is now the laboratory for the Pattern Analysis Branch, NORDA Code 371.

task for the automated cartography environment (lead laboratory status in OCR development*.)

It is important to note that the two parts of the overall automated information extraction and conversion task, i.e., data acquisition/document management and character recognition, must be integrated to produce a "total OCR system" for digitizing M/C analog (graphics/image) documents. Neither software subsystem by itself is a stand-alone deliverable to solve DMA Center production problems in "character recognition."

At the beginning of FY-82, NORDA was tasked by DMA to integrate these two parts to form a total system for computer reading of smooth sheet depth sounding data and to extend this technology to other character recognition problems. Based on these integration efforts, NORDA is scheduled, in FY-84, to deliver to the DMA Centers a prototype system for "digitizing" smooth sheet data. (Brown, 1983b)

1.5 ISOLATED CHARACTER RECOGNITION: CONSTRAINED AND FREE-FORM

Over the past four years, the automated (computer) recognition of free-form, unconstrained, handprinted characters appearing on the M/C source documents has been the target of intense investigations and development at the NORDA Pattern Analysis Laboratory. This character image-to-digital representation (e.g., ASCII code) transformation has required the research, development, extension, and refinement of new "OCR" techniques in order to meet the DMA unconstrained, high-accuracy character recognition requirements. The approach required by this difficult M/C problem is considerably different from that used in constrained, fixed (machine) font OCR systems available from industry**. The algorithms required to capture such M/C information implicit in the varying symbol shapes in which the same character is handprinted are considerably more complex than those for constrained character recognition.

The basic reason for this difference between the constrained and free-form recognition problems is the nature of the "information coding." This unconstrained, coding problem is composed of two parts:

---

*"As a result of the 29 September 1978 IPR, it was unanimously agreed upon that NORDA would become the primary laboratory responsible for OCR algorithm development for DMA." (IPR, 1978) "The IPR highlighted the fact that NORDA is the only laboratory intensively investigating the OCR as it relates to DMA applications." (Macomber, 1977)

**The 1978 IPR determined that OCR systems for unconstrained handprinting were not available (IPR, 1978); See also Overview, (1978).

(1) the meaning (proper label) of a character is not explicitly represented in a fixed geometric form of the symbol but is contained in a complex measure of the relative shape of the various parts of the symbol; e.g., 2, 2, 2. Thus, the symbols of the same class which are to be recognized can range widely in style, geometric detail, and construction, e.g., pen width.
(2) the characters do not appear in a regular or standardized "layout" on the M/C products; e.g., they can vary widely in size, orientation, positioning and spacing; e.g., the orientation can even change within a "word" as in the case of names which are laid out along an extended cartographic feature like a mountain range or river.

The M/C "layout flexibility" makes this free-form problem even more severe than in the case of engineering drawings in which some regularity in the "lettering" is imposed; e.g., engineering symbols are usually constrained to lie horizontally or vertically, with individual character axes parallel to the "word axis," and with constraints on the spacing between the letters.

The M/C free-form symbol recognition problem is complicated by the requirement of high performance, both in accuracy and efficiency, over a wide range of input data quality. To deal successfully with the three factors just mentioned, style, layout, and performance, the PAL has developed a set of sophisticated algorithms for information content/shape measurement and classification/labeling logic for isolated symbols. This Handprinted Symbol Recognition (HSR) System is almost independent of the orientation, size, shape/style of the input data. Furthermore, the system has been implemented in such a manner that the control of its performance is determined by its design:

- the accuracy is basically determined by the recognition logic itself which employs a quality assurance module to guarantee that labeled characters have a very high probability of being properly recognized; i.e., the HSR System can identify characters which are "unrecognizable" with its current decision logic and "rejects" them instead of running the risk of misrecognizing them (undetected substitution error). Thus, ill-formed, rejected characters are flagged for manual recognition/editing.

- the efficiency of the HSR System, that is, the percentage of characters labeled, is a function primarily of the input quality of the data (provided that the character set is one for which the HSR System has been "trained").

1.6 PROJECT DOCUMENTATION, REVIEWS AND MEETINGS

Several kinds of documentation have been generated which describe various aspects of this "Optical Character Recognition Algorithm Development" Subtask including government technical notes,

contractor reports, technical papers, and complete quarterly progress reports. Table I lists this documentation for the project beginning in FY-78. Items 1 and 2 present background information concerning a feasibility study which was contracted by the Naval Oceanographic Office, Hydrographic Development Division with the University of Saskachewan. Item 3 presents a comprehensive review of OCR technology and techniques, as requested by DMA, in the June 1977 IPR. Items 4 and 5 discuss the experimental results of the feasibility study of OCR algorithm (Bolton, 1978) and of the CHITRA algorithm (Dasarathy and Kumar, 1978). The CHITRA techniques were a forerunner of the current work and represent a transition from the work of Bolton to the current HSR System. Item 6 was a significant contract report which reviewed the CHITRA approach and provided material on which the original design of the HSR approach was formulated. Item 8 was an invited paper at a special IEEE session on image processing and presents an overview of the HSR System, Version 1.0. The contract report, item 10, contains various information for this NORDA Technical Note.

The advance development carried out under this Subtask has been the subject of a number of reviews and meetings. Table II outlines these activities and the deliverables prepared under this projects.

TABLE I. Papers and Reports Generated Under OCR Subtasks

1. Bolton, R., "A Cartographic Optical Character Recognition System."

2. Gronmeyer, L. K. and B. W. Ruffin, "An Application of Optical Character Recognition Techniques for the Digitization of Alphanumerics at the Defense Mapping Agency (DMA)--Part I."

3. "An Overview of Optical Character Recognition (OCR) Technology and Techniques."

4. Lybanon, M. and L. K. Gronmeyer, "Recognition of Handprinted Characters for Automated Cartography: A Progress Report."

5. Brown, R. M., M. Lybanon, and L. K. Gronmeyer, "Recognition of Handprinted Characters for Automated Cartography."

6. Gonzalez, R. C., "Evaluation of the CHITRA Character Recognition System."

7. Brown, R. M. and L.K. Gronmeyer, "Recognition of Handprinted Characters for Automated Cartography."

8. Brown, R. M., "Handprinted Symbol Recognition System: A Very High Performance Approach to Pattern Analysis of Free-Form Symbols."

9. Brown, R. M., "Handprinted Symbol Recognition System: A Key Element in Automated Cartography."

10. Cheng, C. F., "Image Preprocessing and Handwritten Character Recognition for Automated Cartography."

11. Brown, R. M. and C. F. Cheng, "Image Preprocessing for Handprinted Symbols for Automated Cartography."

12. Gonzalez, R. C., "Syntactic/Semantic Techniques for Feature Description and Character Recognition."

TABLE II. Meetings/Reviews/Deliverables

	CY78 J0 AN SD	CY79 JAJO FMAN MJSD	CY80 JAJO FMAN MJSD	CY81 JAJO FMAN MJSD	CY82 JAJO FMAN MJSD	83 J F M
IPR, 22 June 1978	Δ					
OCR Overview (1978) Delivered	*					
Kickoff meeting for redirected OCR Algorithm Development Subtask August 1978	Δ					
DMA/NORDA/RADC First joint meetings concerning RSCR Oct 78 (selection of smooth sheets)	Δ					
Review of PAL new facilities and progress (Mar 79)		Δ				
HSR 0.5 (CHITRA) Aug 79 Delivered		*				
Major Progress review Nov. 1979		Δ				
HSR Version 1.0 Delivered Oct 1980			*			
Major Review of Subtask Oct 1980			Δ			
"Handprinted Symbol Recognition System: A Key Element in Automated Cartography" Dec 80 Delivered			*			
Overview of HSR Version 1.0, Delivered "Handprinted Symbol Recognition System: A Very High Performance Approach to Pattern Analysis of Free-Form Symbols" March 81				*		
HSR Version 1.5 Completed Aug 81				*		
HSR Version 2.0 Completed Aug 82					Δ	
NORDA Technical Note 185 Delivered						*
NORDA Technical Note 187 (this report) Delivered						*

2.0 RECOGNITION DOMAINS FOR UNCONSTRAINED HANDPRINTED CHARACTERS

2.1 DIRECT AND DERIVED IMAGE COMPARISON SPACES

The HSR technology developed at the NORDA Pattern Analysis Laboratory has been designed to capture the critical shape information contained in free-form, unconstrained characters appearing on DMA map and chart products. This information content must be such that the recognition/decision logic can accurately and efficiently label or classify characters even though they appear in a wide range of styles, sizes, orientations, and writing instrument construction, e.g., line weight-to-size ratio. Thus, the shape properties which are measured and used by a handprinted Optical Character Recognition (OCR) system must partition the feature-decision space in such a manner as to allow a wide intra-class variation of the characters, yet demonstrate high performance for inter-class separability and recognition. Determining what shape features to employ for this difficult partitioning problem and how to measure them explicitly has been a key area of research in the field of unconstrained handprinted character recognition.

The basic transformation process that converts (visual) character images to computer compatible codes properly describing the character image can operate in two different domains:

(1) the direct image space domain made up of the black and white points of the binary raster scan of the character;

(2) a derived feature measurement space domain obtained from the image space through some computer process.

The character image space is basically what is provided by the raster scanner. In this space, an isolated character is represented by a matrix of black and white points in the form of a digital (sampled), thresholded image. The characters range in height from 50 to 100 samples; the widths are comparable. Thus, these binary (black/white) image matrices typically range from about 48 x 64 to 96 x 128. The size of the actual digital image is directly determined by the scanner resolution and the original analog (graphic) character size on the M/C product.

The sampling interval or resolution must be set so that all necessary shape properties of the character on the graphic document are recorded in the digital raster scanned image. It has been found that a sample interval equal to 2-3% of the size of the numeral size on the original document generates reliable digital images for smooth sheets. At this sampling interval, a minimum of three sample points are obtained across minimum line weight lines or across narrow openings which contain or convey the shape of the image. This requirement guarantees that the sampling will not cause "drop outs" on thin lines or fill-in where two lines or characters are close together. Such character matrices can be seen like other digital images in either hard or soft copy. When viewed

in this manner, the handprinted lines have a definite size or thickness.

A derived feature measurement space is obtained through some transformation or measurement based on this direct character image space. The search and definition of such derived feature spaces has been a major task in handprinted character recognition under this Subtask and will be discussed at length in Chapter 4.

How these two domains are used in OCR systems will be discussed briefly in the remainder of this section. Simple template matching schemes make their decisions or recognize characters on the basis of comparisons in the image space domain. "Image templates" of the targeted character set are entered into the system "to train the recognizer"; the recognizer then converts the incoming unknown characters to their computer compatible codes on the basis of matching their direct image space properties with these stored templates. To handle a wide range of styles, sizes, orientations, and constructions, such an approach requires a very large set of comparison templates. Furthermore, the time required to make all the comparison tests becomes very large.

Such "whole-body" image templates range from point-by-point comparison templates for a direct correlation approach through variations based directly on the point image space; e.g., projection histograms on the x and y axes, the number of points in various predefined sectors of the image, etc. It can be seen that such techniques face at least the problems of (1) size normalization, (2) standard orientations (orientation normalization*) and (3) highly sensitive to minor variations in style and writing instrument, e.g., line weight, etc.

For fixed-font character recognition systems where these problems do not arise, the image space comparison approach has been highly successful. Even in these cases, however, quality control procedures must be maintained to assure repeatability of the character images; e.g., new ribbons, good quality paper, and careful setup are required. Without such controls, the performance deteriorates rapidly. Extension of these concepts for constrained handprint has had only moderate success, again, the recognition accuracies for such approaches have not met the DMA map and chart requirements (near-zero undetected substitution errors).

Attempts to remove the restrictions imposed by the use of direct image comparison spaces have led to the development of derived feature spaces that are more and more abstracted from the direct character image. The histogram and sector density approaches mentioned above are first steps in this direction; however,

*Standard orientation is very difficult to define since it must properly deal with skew and style as well as rotation.

they depend explicitly on the "details of the shape" of the character. In general, the goal in defining a derived feature space is to develop measurement techniques that will capture the information content or meaning of an unconstrained handprinted character independent of the specific details in which it is presented to the scanner; i.e., size, orientation, style, writing instrument construction. Examples of derived features include measurements of such things as the number of enclosed regions and the angles which various parts of the character make with one another.

Extensive literature reviews (Overview, 1978) and various experiments and studies at NORDA have led to the conclusion that the DMA requirements for wide variations of unconstrained handprint with high recognition accuracies can be met only through the more sophisticated approach of derived shape/feature measurement techniques (Brown, et. al., 1979; Gonzales, 1980).

The NORDA Pattern Analysis Laboratory has developed two basic approaches to the derived feature space problem and shape measurement; these developments have proceeded along the following lines:

(1) a heuristic approach which attempts to model directly the human recognition process, and

(2) an abstract approach in which formal measurements are made on the character image.

The first approach is based on the concept of the "stroke representation" of a handprinted character. The second approach is based on identifying unique but rather abstract properties of the boundary of a symbol; i.e., the "sides" of the thick line character image. Both of these approaches perform a transformation on the original character image space based on different models of handprinting. The first model requires a thinning or skeletonizing process to reduce the image of the handprinted character to a string of connected single points or "thin line." The second model requires the extraction of the external and internal boundary lines (or "sides of the stroke lines") that make up the image of the character. Thus, both techniques transform the two-dimensional character image into a derived representation space of a one-dimensional string of points.

The first method based on "strokes" has been highly developed and is incorporated in the current HSR System as the basis for shape feature measurements. This technique is discussed in detail in following sections. The second method is still in the development phase and will be important for extensions of OCR technology to symbols that are inherently not "thinnable," that is, which are not generated by simple pen strokes. This method is also being explored as one of the techniques for handling the problem of connected or touching characters that cannot be individually isolated. Finally, it appears that a hybrid approach using both techniques may provide significant increases in throughput speed. This

second method based on boundary lines, will not be discussed further in this report; however, it is planned to present these results in a future NORDA Technical Report. Both of these methods developed at NORDA have extended the state-of-the-art in the field of information and feature measurement for unconstrained handprinted characters.

2.2 HEURISTIC MODEL FOR HANDPRINTED CHARACTER RECOGNITION

Humans appear to handle character recognition at two levels, at least as described in the two extreme cases presented below. When reading a book for example, little attention is paid to the details of the individual characters on the printed page. (1) This mode of character recognition can be called the gestalt mode. However, as one moves from such a highly "structured and familiar reading environment" to one in which more interpretation is required, e.g., in which noise or stylistic variations are present and the context of adjacent characters is not available, then humans appear to begin employing quantitative decision/classification rules based on careful observations of the shape properties of the character. (2) This mode can be called the "code deciphering" mode. Thus, when humans are asked to label handprinted symbols, out of context, that have a wide range of style characteristics, they apparently have a set of criteria that they apply to determine whether the symbol has the necessary properties to be labeled this or that particular character. This second mode represents the complex end of the human character recognition spectrum and might be used as a heuristic model for computer automated recognition of M/C symbols.

This heuristic model leads to a consideration of how characters are constructed and how they convey information; several hypotheses can be made on this basis: (1) The information content needed to convey or recognize an isolated symbol is contained in an "idealized stroke" representation (lines with no width*) of the character image. (2) The set of test criteria used by humans in the code deciphering" mode consists of rules to determine whether the strokes display key geometric and topologic properties for any given character. (3) Feature extraction and recognition algorithms can be developed to measure these geometric and topologic properties (feature vector components) which model the human "code deciphering" procedure. In particular, if a human is given the results of these measurements without seeing the character, then he should agree with the label that the recognition program generated; that is, that the character has a sufficient set of properties such that, if humans tried to "reconstruct the character" on the basis of the given measurement properties, they would be "assured" of arriving at the character labeled as generated by the

*The fact that it is necessary to over-sample the analog graphic is an artifact of the scanning process and is required to guarantee that one can indeed recover the "true strokes" for the character without dropouts or fill-ins in this linear (stroke) representation.

recognizer. This concept is called the Sufficient Class Membership criterion (Brown, et al., 1979). (4) These hypotheses represent a specific guide (1) for determining the proper shape features to measure and (2) for developing the necessary labeling mechanism or recognition logic (decision tree).

These concepts have proved very useful in the development of the stroke model for isolated characters and in the design of the HSR System. Its high degree of success is attributed to the fact that the model appears properly to mirror the "communication channel" used by humans for handprinted information. In fact, the model can even be used to "restore" faulty or filled in (noise corrupted) characters so that the necessary shape information can be regenerated and the character properly identified; see Figure 3.

The success of this stroke concept for feature space/information extraction is dependent on the "accuracy" of the transformation from the sampled image space (i.e., raster scan digital image) to an abstract stroke space of "Euclidean lines and curves." In this context "accuracy" refers to the fact that the transformation must properly represent the original, human-intended, information-carrying features of the character. The development of this fundamental stroke concept (Brown, 1981a) and its incorporation in the HSR System has led to the high performance required for the DMA map and chart character recognition task. The actual thinning transformation process is considered a preprocessing function relative to the HSR System. The details of these algorithms will be presented in the NORDA Technical Note 210, "Image Preprocessing of Handprinted Characters for Automated Cartography."

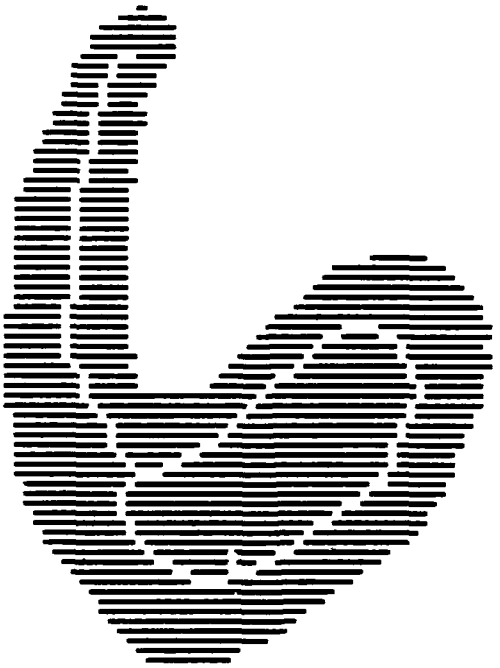


Figure 3. Reconstruction of the strokes in a filled-in "6"

3.0 THE MAP AND CHART MANUSCRIPT ENVIRONMENT

3.1 CHARACTERISTICS OF MAP AND CHART HANDPRINTING

This chapter reviews briefly two topics concerning the map and chart (M/C) environment from an OCR point of view: (1) the characteristics of handprinting on documents used in M/C production and (2) the DMA performance requirements of accuracy and efficiency. Under the first topic several items will be considered:

- general nature and quality of M/C manuscripts;
- line weight characteristics;
- size, orientation, and shape/style;
- broken or disconnected characters;
- connected characters.

This discussion will deal with M/C graphic characteristics only as they affect the handprinted symbol recognition task. The following characteristics related to the overall information content and transformation will not be considered:

- word composition;
- symbol-to-cartographic feature referencing;
- information contained or presented in graphic positions and orientations;
- the use of color, fonts, line types, etc.

In general, cartographers use high quality materials in the preparation of M/C manuscripts; e.g., black ink on mylar. Furthermore, because of the nature of the information presentation job and the training of the personnel involved, such documents are printed with reasonable care. Thus, although these graphics can be free-form and unconstrained, they do not have the extreme range of printing styles and materials that are exhibited in handprinting by the general public; i.e., unusual colors of ink and paper, artistic flourishes, or poor character formation.

The line weight used to print characters on some documents, particularly smooth sheets, introduces a property of the characters that is not always present in other handprinting and which must be carefully treated. Line weight is important because it can affect the ability to reconstruct the strokes of the character. This effect can be measured in two ways: (1) the ratio of the area occupied by the ink lines in the character to the area of a "box" which bounds the character; and (2) the ratio of the line weight to some linear measure of the size of the character. If either of these ratios is significant, then it often is not possible to easily "thin" the characters; see for example Figure 4. In some cases, the large line weights even cause significant information-carrying shape features to be obliterated; e.g., enclosed regions, bays or open regions, etc.; see Figure 5. Although humans can often decipher such characters by recognizing that such properties are implied by the boundary of the "filled-in" character, computer

Figure 4. Poor ratio of line weight to size

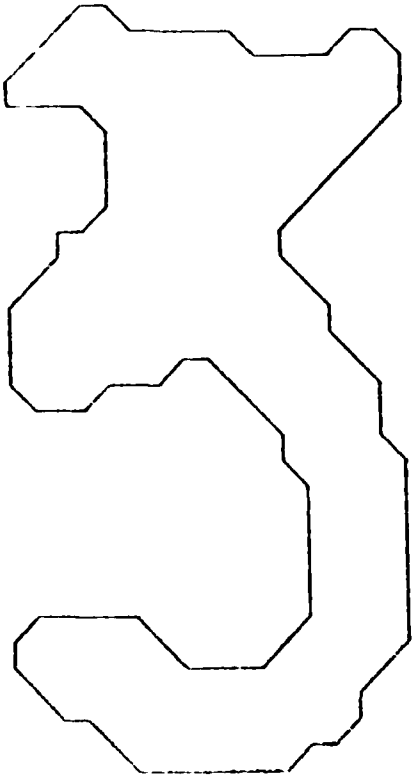


Figure 5. Obliterated opening in the numeral "3" by a line weight which is too thick

algorithms to model this complex human judgment are in the development phase. Their application to solving the line weight-to-size ratio over large and varying databases has not been thoroughly tested. When used for smooth sheet symbols, they have made a significant improvement in the preprocessing and have resulted in higher efficiencies (fewer characters being labeled as unrecognizable).

As will be mentioned repeatedly in this report, the guidelines for the OCR algorithm development require that the recognition techniques be independent of the size, orientation, and style/shape of the characters. These variations have been observed to occur quite often on smooth sheets and other handprinted documents: for example, some documents have multiple authors; size and orientation are often governed by the constraints of where the "data" must be placed on the manuscript; writing instruments can also vary. The important point to note about these handprinting variations is that they all can occur on the same graphic manuscript in essentially an unpredictable manner.

Handprinted OCR algorithms are based on recognizing isolated characters which are then composed into "words." More complex systems to identify or recognize "words" as "whole units" are completely outside the scope of this Subtask. An isolated character approach requires that the whole character and nothing but the single character be presented to the recognizer. In generating isolated "cartographic objects" on an M/C manuscript, raster scan character acquisition algorithms, developed to date, examine the connectivity of the black points on the document (e.g., the overlap of run length codes). This approach gives rise to the concept of "connected components," since these algorithms can easily locate and extract "symbols" that are all connected together by "ink dots." The problem arises, however, because even on the original manuscripts, these "connected component symbols" are not always "complete characters." A typical example from the numbers is the numeral "5". Many people construct this numeral with one pen lift during its printing. This "second stroke" is used to "put the top (horizontal bar) on the character." When sufficient care is not taken, however, the top bar can be disconnected from the main body of the numeral. Such examples of disconnected characters, are even more common in the capital alphabet; e.g., "I", "J", "E", "F," etc., are often printed in disconnected parts.

The recognition of these characters by a human observer is usually quite easy unless the words or characters are very badly distorted. In these easy cases, humans appear to perform the character isolation and recognition "all in one step." More sophisticated symbol extraction algorithms are under development at NORDA which should be able to compose the "broken characters" into "isolated whole characters," at least in the cases where the inter-part separation is smaller than the inter-character separation. A further constraint can be added which will minimize composition errors; namely, the requirement that all "recombined characters"

must result in the fact that each "new character is properly recognized by an augmented recognizer. This approach still requires a two-step process that only partially models the parallel human observer.

The last significant characteristic of M/C manuscripts is the occurrence of connected characters. Again, this problem is caused by the fundamental isolation process during data acquisition. The situation for connected characters, however, can be even more severe than in the disconnected case. Broken or disconnected characters are basically a "local" phenomenon; i.e., the parts of the character are contained within an area approximately the size of regular characters. Indeed, it is this property which allows a reconstruction procedure to be developed. In the case of connected characters, however, several situations can arise:

(1) Two adjacent characters can touch and therefore occupy more space than a single symbol.
(2) Several characters can touch each other directly in a chain.
(3) Any number of characters can be connected by "lines" which are not part of the characters in the string; e.g., contour lines, coffee stains, smudges, or even scanner artifacts.
(4) Single characters can be connected to non-characters (trash).
(5) Both the broken and the connected character problem can be present at once. For example, it is not uncommon for the top bar on a numeral "5" to be disconnected from the main body of the "5" and, furthermore, to be connected to the adjacent numeral on its right.

Algorithms are under development at NORDA to handle case 1 above. One can easily see, however, that the other cases are considerably more complex. Fortunately, with the exception of the single part disconnection/connection case (e.g., the number "5" just mentioned) these "higher order" connected characters occur less frequently on smooth sheets. Also the "5-type" problem may be sufficiently "local" to be treated under case 1. The current work on connected characters looks promising. The guideline that each "character" recreated by the disconnection algorithm must be recognizable should maintain the accuracy of the system.

3.2 PERFORMANCE MEASUREMENTS

In an automated cartography environment, computer algorithms must exhibit a high level of performance. Such techniques are often compared to human performance for the same task. This situation, however, is not the prime driving factor for the automated performance levels; rather, it is the fundamental M/C product requirements that set the performance whether by man or machine. Indeed, it has only recently been possible that some AC processes

could be performed by computers with levels of performance approaching that of humans. Automated recognition of unconstrained handprint is in this category.

Three terms related to performance are discussed in this Section: ground truth, accuracy, and efficiency. In discussing unconstrained handprint OCR performance of advanced development systems, it is important that these three performance concepts be understood and defined. In the context of an OCR experiment, the term ground truth (GT) is the name given to the human-identified "value" for the handprinted "information" on the M/C graphic document. This term is borrowed from the satellite image domain where an experimenter collects data by making measurements at the "ground site" appearing in the satellite image so that he can compare these "known ground truth values" with those generated by his classification algorithm.

From a production point of view, the end goal of a "total OCR system" for unconstrained handprinting is to extract information from a manuscript with as little human interaction or assistance as possible. Therefore, the bottom line in performance must be gauged by this standard. When the information is contained in "words" on the M/C graphic document, then a performance parameter must measure the success of "digitizing" the information in such "words." In the case of smooth sheet data, for example, these "words" are sounding values made up of several numerals or digits. Thus, from a "production digitizing" point of view, the ground truth is the single numerical value of the depth value of the sounding. From a character image-to-digital code transformation point of view, however, the ground truth is the "meaning or value of the isolated numeral" in the raster scan which is assigned to each digit of the sounding image.

This difference in point of view has two consequences. First, the ground truth labeling of isolated characters out of context is a more difficult task for humans than the "regular digitizing" problem. As a result, one must exercise considerable care in obtaining ground truth and in setting up performance evaluation experiments for developmental testing of an OCR scheme. One must remember that the recognition algorithms themselves do not use context; consistency checks on the final product which use the "meaning of the words" are not considered part of the OCR algorithms for map and charts at the current level of development.

One reason that care must be exercised is that humans may assign a different label to characters out of context from those identified in-context. Furthermore, different people may not agree on the label for ambiguous characters. However, from a production point of view, these distinctions are irrelevant.

The second consequence is the "multiplication effect" between the performance for isolated characters and for "complete words." To illustrate this effect, consider four-digit soundings on smooth

sheets. A substitution error rate of 1% in isolated numeral recognition can translate into a 4% misrecognition rate for sounding values. Thus, 400 numeral images are required to make up 100 four-digit soundings values. A 1% error rate at the isolated character image level would mean that four numeral images or digits were incorrectly labeled. If these errors are distributed in a worst case manner, four soundings would be affected and the error rate at the sounding value level would be 4%. This inherent multiplication effect between characters and "words" imposes very high performance requirements on an OCR system.

Continuing the smooth sheet example just considered, if one requires a 99% accuracy level for sounding values, i.e., 1% undetected incorrect values of depth, then one must recognize isolated numeral images at a 99.75% accuracy level (0.25% undetected substitution errors).

It appears, in general, that this multiplication factor does not affect human recognition in the same manner as OCR systems, since humans usually use the gestalt mode of recognition and context information. Indeed, error rates in a "total OCR system" could be made less than the implied rate of the isolated character recognition algorithms employed if the total system uses a context checking scheme which models the human judgment concerning the information from the surrounding region on the manuscript. Thus, such a total system would employ several stages to complete the overall information extraction process including a recognition algorithm followed by a context algorithm.

A straight forward use of such context information results in some "words" being rejected as "not making sense in the overall sentence structure." In the smooth sheet example, the "sentences" are the "geophysical contour possibilities" for the sea floor; these, as well as other abstract M/C "information words," are certainly more difficult "to read" than regular sentences. Expert cartographers, however, display sophisticated judgment when reviewing M/C documents which usually allow them to detect "word discrepencies," e.g., "valid shoals values" as opposed to "false shallow depth spikes." Context checking in an OCR system, however, results in a lowering of efficiency (the percentage of correct "words" recognized) since the whole word must be rejected; the use of context to correct individual misidentified symbols requires a considerably more complex approach.

In the examination of the two points of view about evaluating OCR systems, two measures of performance have been introduced: accuracy and efficiency. Furthermore, one can easily see that these two properties of a system can represent competing requirements; that is, if one wants high efficiencies (every character image automatically being assigned a label), one may run a higher risk of introducing undetected substitution errors. The HSR System development discussed in this report prioritized these two performance characteristics. The system was first optimized for accuracy; then attempts were made to increase its efficiency without loss in

accuracy. This approach has basically been successful. As indicated in Section 1.5, this approach led to the development of a design which separated the control of these two performance properties.

Returning to the practical matter of developmental testing of an OCR system and properly ground truthed testing sets, several issues need to be considered: (1) ground truth problems and human recognition performances; (2) size, accuracy and complexity of test sets; and (3) time and experiment run problems. The problems of ambiguous characters and ground truthing in and out of context have already been mentioned. There is also the complex technical matter of whether ground truth should be assigned to the raster image or to the thin-line ("vectorized") image sent to the recognizer. During the HSR development this issue was handled by using both kinds of ground truth. A comparison of the raster image and stick-figure (thin-line) ground truth lead to significant improvements in the preprocessor which generates the inputs to the recognizer. Comparison of the stick-figure ground truth with the recognizer outputs allowed careful analysis of the HSR performance. Now that the performance of both the preprocessor and recognizer have achieved high levels, figures of accuracy and efficiency are determined by the comparison of raster image ground truth to recognizer output values unless otherwise stated.

The ground truth for the PAL data sets was assigned to each individual symbol viewed out of context from the other symbols that may have been in the surrounding region on the document. In the analysis of ambiguous symbols, reference was sometimes made to the original graphic document. In general qualitative terms, it was found that the HSR System had a higher performance than the human ground truther. This statement, which appears at first as a contradiction in terms, means that a comparison of the disagreements between the ground truth value and the recognizer value often resulted in the fact that the human either mistyped the entry, or in some cases, apparently mislabeled the character. This second determination was made by comparing several "double-blind experiment" values of different people for the symbols in disagreement. (See also the analysis of human performance for out-of-context symbol recognition reported by Niessen, 1960.) As was demonstrated in the HSR testing, the high performance levels being measured require one to construct the data sets very carefully because one is approaching the limits where "human recognition errors" can contaminate the problem; i.e., without considerable care, the OCR algorithms perform better than humans on <u>isolated</u> numerals.

The final technical problem to be considered is the large, completely ground truthed test sets that are needed. A simple analysis indicates that approximately 100,000 to 200,000 symbols are required for adequate testing of a recognition algorithm for the numerals alone. The range in the data test set size reflects the range of actual targeted numeral pattern categories in question (See Section 4.9). If one anticipates an approximate substitution rate of 0.3% for a given pattern category, then one needs

10,000 symbols in order to generate 30 errors in that category. Statistical guidelines indicate that these numbers would allow valid statements to be made about the measured error in the substitution rate. If there are between 10 and 20 target pattern categories (since some numerals have more than one unique pattern shape, the numeral "4", for example), one then arrives at the data test sizes indicated above.

Ground truth data sets of this size involve considerable amounts of labor to establish accurately. Furthermore, exercise of an OCR algorithm to determine various threshholds, parameter settings, feature measures, and overall performance over these large test sets is quite time consuming. As a consequence, the developmental testing at the PAL has employed smaller sets initially involving approximately 7,000-10,000 isolated properly scanned numerals. Based on these tests, the measured values of performance for the HSR System, Version 2.0, can be stated: an accuracy of 99.7% (0.3% substitution errors) and an efficiency of approximately 95% on clean smooth sheets. FY-83 plans call for more comprehensive tests as the database grows to over 100,000 symbols.

4.0 THE HANDPRINTED SYMBOL RECOGNITION SYSTEM

4.1 HSR SYSTEM STRUCTURE

This Chapter discusses the general structure of the Handprinted Symbol Recognition (HSR) System, Version 2.0. Five major functions are performed by the HSR System in transforming the isolated input character image to the derived feature space of geometric and topologic measurements and then in assigning the input character a computer compatible code based on these shape properties. These basic functions are:

(1) Segment generation

- list structure
- linear approximation

(2) Artifact removal

- spurs
- improper crosses

(3) Curve analysis and measurement

- stroke generation
- geometry measurement

(4) Shape feature extraction

(5) Decision tree processing or recognition

Each of these functions is described in the following sections of this Chapter.

Before proceeding with these descriptions, however, several observations should be made about automatic image classification and handprinted character recognition in particular. Any recognition system is basically made up of two parts:

- feature extraction
- classification scheme

Extensive efforts have been reported in the literature concerning various classifiers and recognition logic (see Overview, 1978). This situation is probably due to the fact that the analysis and development of this part of an overall recognition system is quite amenable to detailed mathematical treatments of various kinds. Only limited research has been found dealing with feature extraction for handprinted symbols; i.e., research on the adequate measurement of the shape of characters (e.g., Suen, et al., 1980; Blesser, et al., 1976; Pavlidis, 1980). Again, this situation follows from the fundamental nature of the character description problem and the difficulty in determining the "information-carrying structure" or features for unconstrained handprinting.

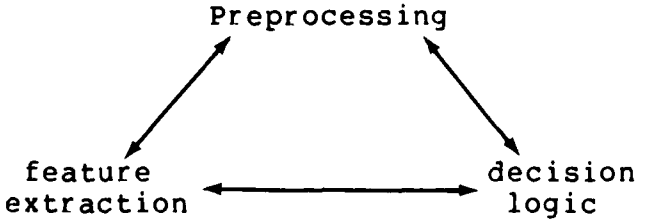
Review of the literature and experience at the NORDA PAL has indicated that this second area, shape description, is the fundamental key to the advanced development of an unconstrained character recognition system. A solution to this difficult problem involves a careful blending of heuristic approaches, appropriate abstraction of these and other measurements concepts, algorithm development, and extensive testing on large data sets of actual (realistic) handprinted images.

A number of algorithms and systems for handprinting have been reported, many of which cite high performance figures. A careful review of these techniques reveals, however, that the test data involved usually has several flaws: (1) often, the images are synthetically generated by hand on a grid matrix; (2) the handprint is constrained; (3) the data sets are small and do not contain a large enough range of significant "problem characters." Thus, the comprehensive problem of character recognition for images unconstrained by size, orientation, authorship/style has received only limited attention.

Finally, it should be noted that there is an important interaction between the preprocessing of the character images and the recognition system. In particular, if the images have little or no preprocessing, then the feature extraction or shape measurement part of the recognition system can often be misled; e.g., it can easily generate descriptions that are contaminated by noise artifacts in the image. These noise contaminated descriptions can lead to mislabeling by the recognition logic. To avoid such false descriptions, the feature extractor must be very sophisticated and, in effect, incorporate "preprocessing-like" functions. However, to develop a very high performance preprocessor requires various shape measurements to be made during the preprocessing function (NORDA Technical Note 210, in prep.).

Similar statements can be made about the relationship of the feature extractor and the decision logic of a system. If one uses a rather simple feature extractor that generates crude character descriptions, then the decision logic or classifier must "work very hard" in sorting out the correct label for the character. Conversely, a "very smart" feature extractor can almost become a recognizer in its own right, making the decision logic almost trivial.

The NORDA Pattern Analysis Laboratory has extensively investigated each of these interactions:



This comprehensive approach to the total unconstrained character recognition system has led to an optimization of each of these functions in relationship to the others. As a result, important advances in the state-of-the-art in high performance character recognition for unconstrained symbols has been possible and the ground work for automatic map and chart reading has been laid.

4.2 SEGMENT LIST GENERATION

The input to the HSR System, Version 2.0, is a preprocessed raster image of the isolated numeral to be recognized. The preprocessing stage has already converted the direct character image to its "thin-line" representation. Figure 6 shows an example of this thinning process. This thin-line representation, however, is still in a raster format; it can be seen in either hardcopy or softcopy as a "stick figure image" made up of connected sample points in the form of lines with "zero thickness." More important is the fact that this thin-line character image has not been organized to give it the necessary "stroke structure" for further processing. The data description of this input image is a simple raster pattern of picture elements (pixels) or a more compact version of the raster format; namely, a list of all the points that are black starting with coordinates in the upper left corner and proceeding along each scan line from left to right and from top to bottom.

The first function performed on this input data is to increase its information density content by organizing the image points into segment lists. To perform this function, each point in the input stick-figure image must be classified on the basis of the points in its immediate 3 x 3 neighborhood; its connection to these neighbors must be determined; and its ordering in the segment string must be assigned. This segment generation process is based on the 256 possible 3 x 3 matrix neighborhood patterns; any raster image point can be classified as one of the following types:

(1) isolated points have no neighbors;
(2) end points have only one neighbor;
(3) "regular line" points have only two neighbors;
(4) branch points of type three have three neighbors;
(5) branch points of type four have four neighbors;
(6) any higher connectivity or points with neighborhoods containing too many "closely packed" points are defined as "improperly thinned"; thin-line images containing such configurations should not be generated by the thinning preprocessor; therefore, this category generates an error message.

Examples of the 3 x 3 matrices used in this classification scheme for thin-line image points are shown in Figure 7.

An image segment is then defined as a connected, ordered set of points bounded by two terminators; a terminator is an "end

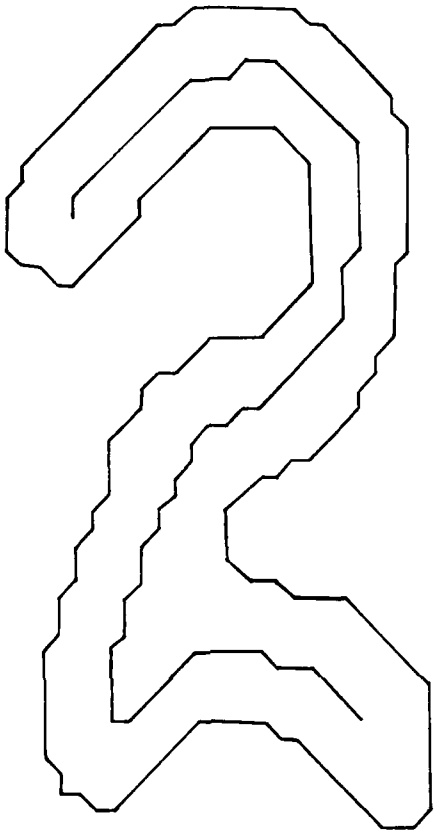


Figure 6. Example of good thinning

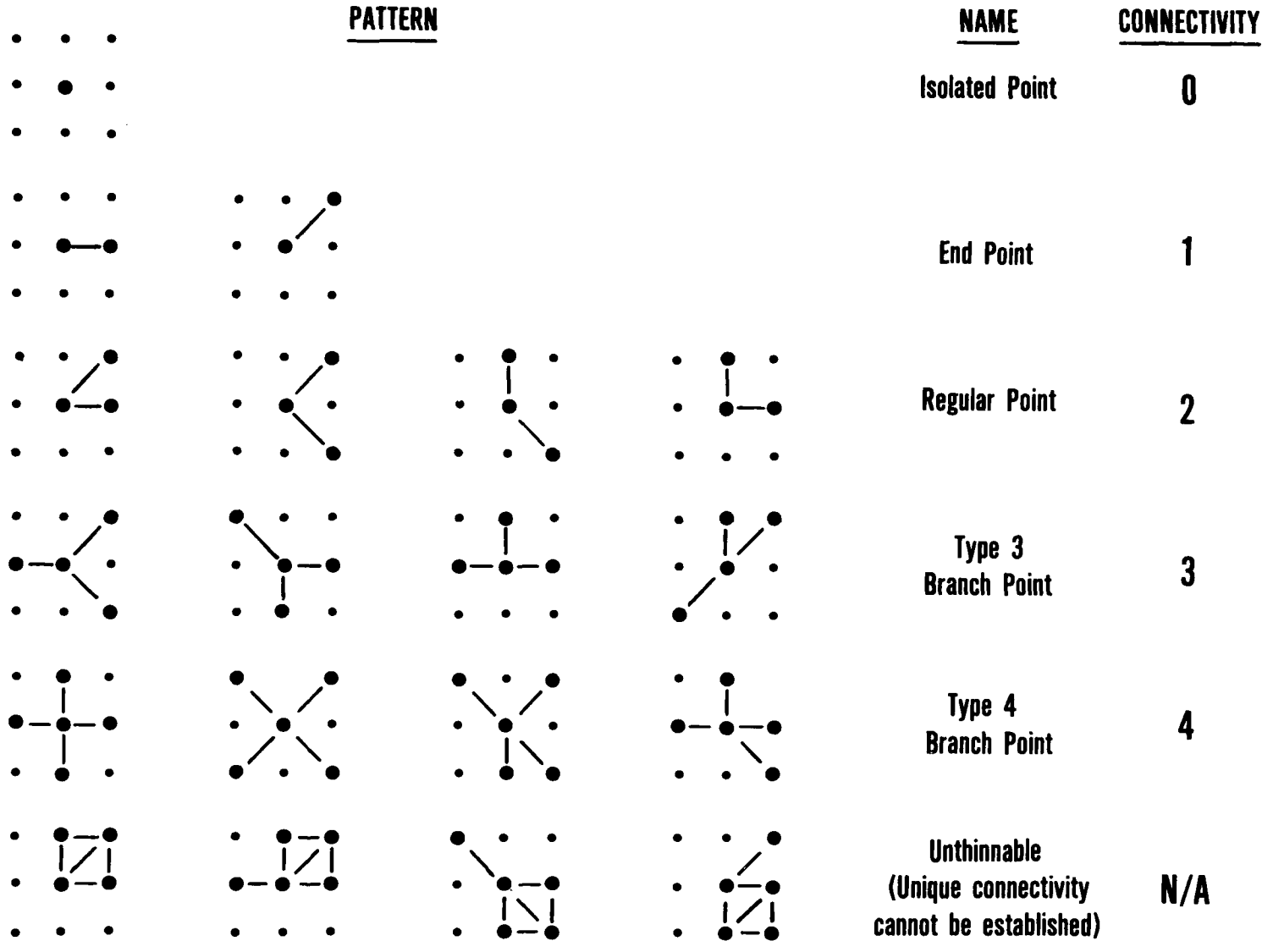


Figure 7. Examples of 3x3 connectivity matrices

point" or a "branch point." Each segment must have at least two points; single segments enclosing a region are a special case in which the beginning and ending terminators of the list are the same point. The result of this segment organization process is a derived information structure for the character made up of an ordered list of the points in each segment of the character and a segment description list containing the top-level information about each segment; e.g., its terminators, number of points, etc. This information density increasing process is relatively time consuming but it is central to further HSR processing. On the basis of the list structures so generated, one can build the complete "stroke structure" of the character. This list processing approach is a major advance over earlier work in handprinted character recognition.

4.3 LINEAR APPROXIMATION REPRESENTATION

The process of increasing the organizational information content by getting the segment structure of the character is followed by a processing module which generates a linear approximation of the segmented image. This process is made up of three subfunctions:

(1) it acts as a filter to remove localized (single point) variations in the image,
(2) it compresses the character description by reducing the number of points in the segment lists, and
(3) it identifies special extreme points.

Several linear approximation schemes have been implemented and tested at the PAL. Their basic function is to select from the input segment list structure a subset of points which adequately represents the character. An example of the filtering generated by this selection and compression process is shown in Figure 8. In the original raster input image, or in the segment lists, the lines joining adjacent sample points are restricted to intersect at multiples of 45° and their lengths are restricted to 1 unit or $\sqrt{2}$ units. Therefore, single noise points can cause a local high geometry change. For example, Figure 8 shows how this local phenomenon affects the "beginning angle" of the segment at the point marked "A." This figure also indicates the distortion in the "turning angle" at the point marked "B." Figure 9 shows a linear approximation representation for this case.

The linear approximation technique which appears to give the best results is a global, recursive binary search for the points that are most distant from the line joining the end points of the segment and its subsequent partitioned subsegments. This process is illustrated in Figure 10 and is discussed further in Cheng, (1983). The process halts when the maximum distance becomes less than a pre-defined threshold. Excellent results have been obtained with this algorithm; furthermore, such an algorithm appears to be essential to obtaining proper curvature measures for the character.

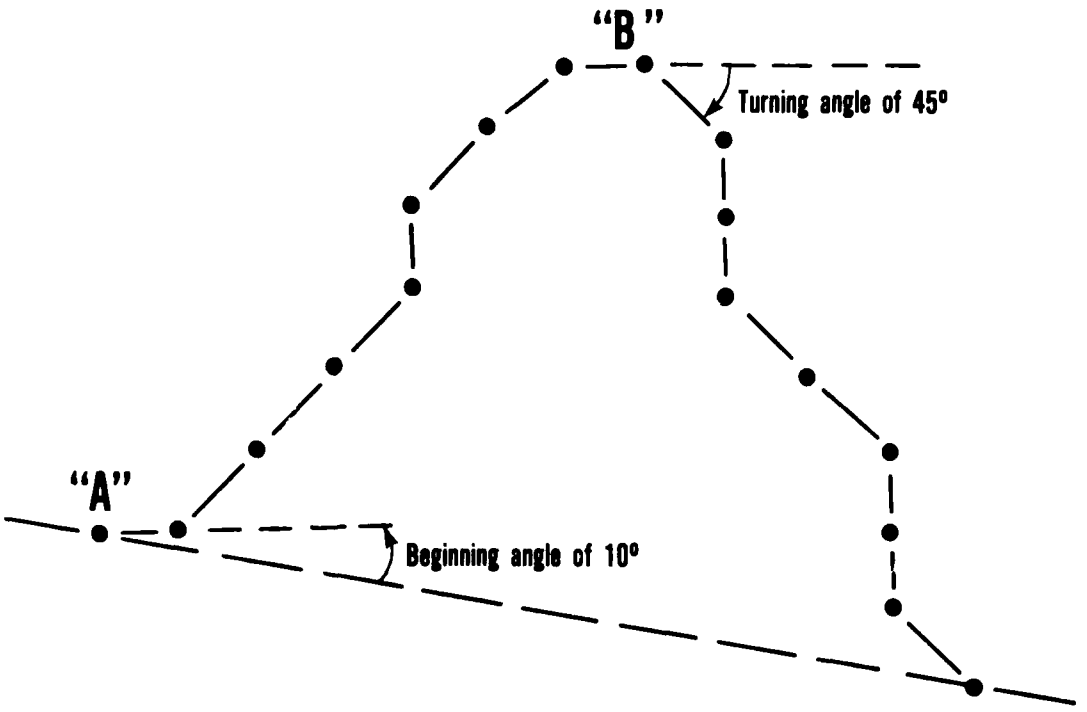


Figure 8. Distortions in beginning angle and turning angle when no approximation filter is applied

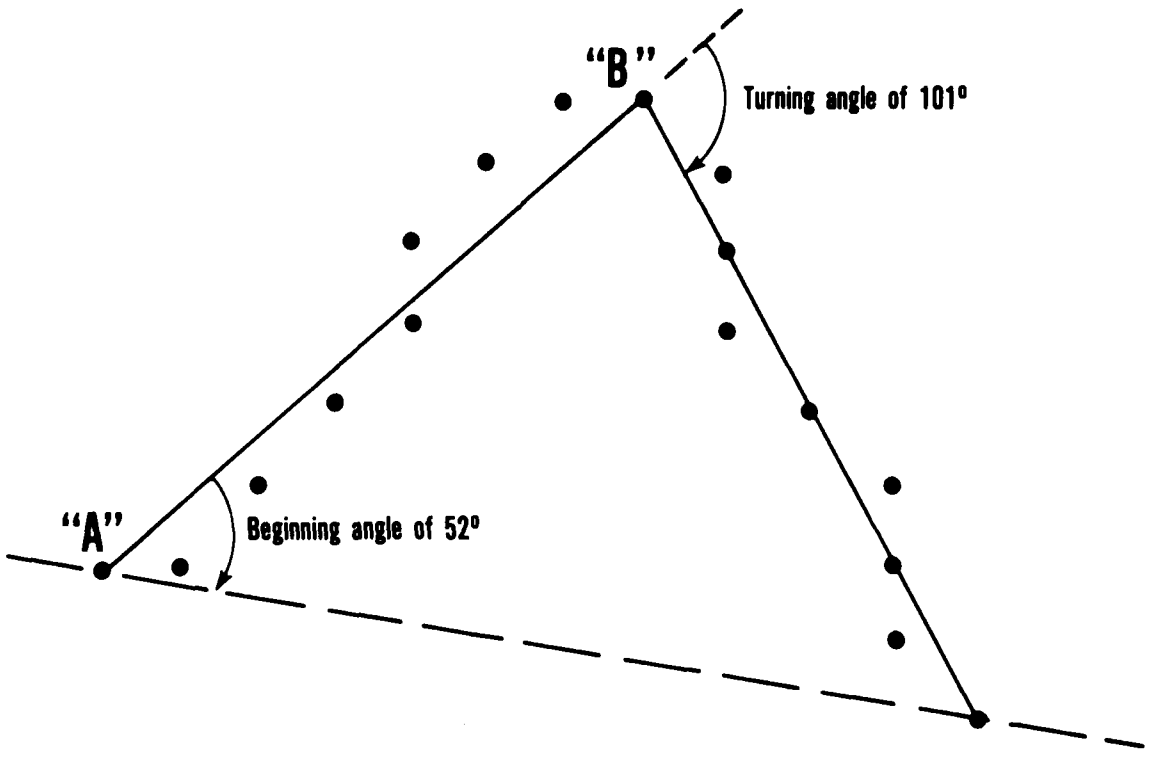


Figure 9. Linear approximation removes local geometric distortions

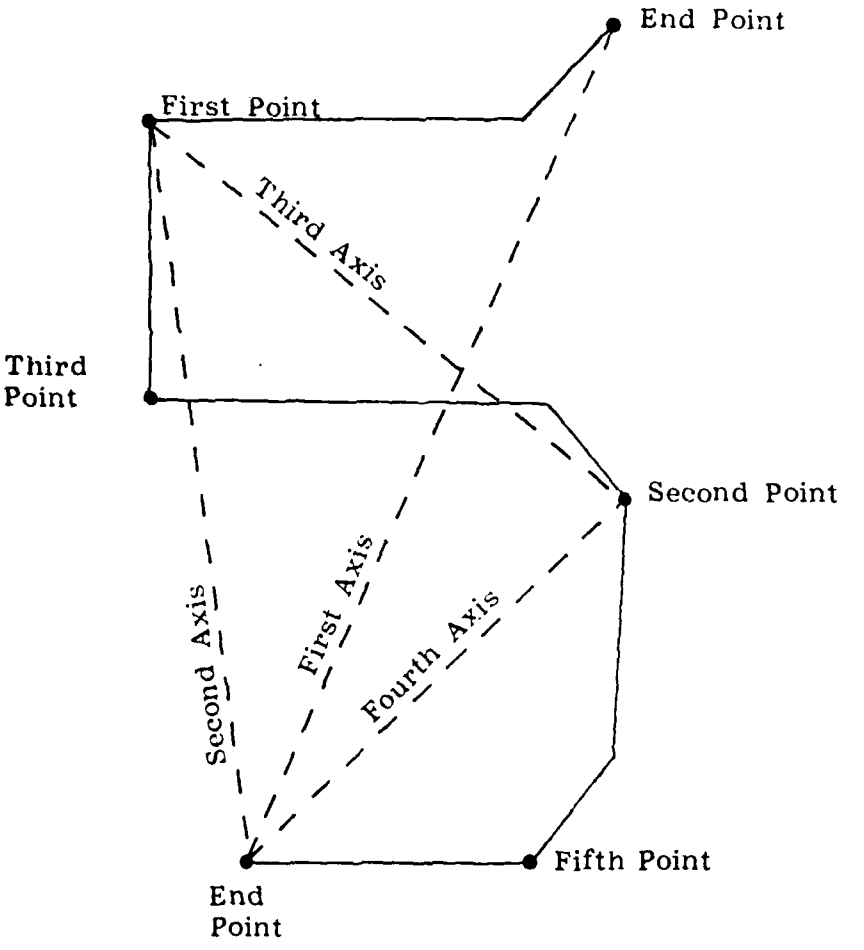


Figure 10. Recursive binary search for approximation points

Two earlier approximation techniques were used which are examples of forward looking, non-global, single-pass algorithms; an approach suggested by Gonzalez (1980), and Gonzalez's approach as modified by Brown. These techniques were not as satisfactory as the binary search algorithm relative to three criteria:

- Their approximate representations showed a higher rms deviation from the original thinned figure image than for those generated by the binary search approach.
- They did not maintain the intuitive "critical points" or extremas of the characters.
- The representations had a tendency to produce a higher degree of smoothing or rounding at the corners of the characters.

The numeral seven ("7") in Figure 11 illustrates this rounding effect. The main advantage of these earlier algorithms was their speed.

From one point of view, the linear approximation module transforms the input character image by deriving a more abstract representation. However, this representation, in some respects, is more similar to the original handprinted stroke form of the character on the graphic document. This representation can be thought of as no longer consisting of discrete sample points on a square grid matrix but as made up of "continuous" straight line segments joining the approximation points. The angular resolution between such line segments is no longer (so greatly) restricted; likewise, the length of such approximation line segments is also less restricted. These important measurements of angle and length are used in defining fundamental shape feature descriptions of the input characters. Finally, the PAL has performed limited experiments in which the "continuous curve" joining the approximate points was more complicated than a "simple linear fit"; in particular, cubic splines have been investigated. These "higher order approximations" built on this approximation representation of the character; they are still experimental in nature, however, and have not been included at this time in the HSR System, Version 2.0.

4.4 SKELETONS AND OTHER ARTIFACTS IN THE CLOSET*

As mentioned in Section 2.2, the accuracy of the preprocessing operation is critical to proper geometric and topologic feature extraction. The thinning or skeletonizing preprocess which generates the input data for the HSR System can create artifacts in these thin-line input images. Three specific examples of "skeleton failures" in the thinning process are shown in Figures 12, 13, and 14: these artifacts are called (1) spurs, (2) artificial mid-points for stroke crossings (improper crosses), and (3) displaced branch points, respectively. These artifacts are particularly easy to identify after the segment list structure and the

*Compare Hilditch, 1969, "Linear Skeletons from Square Cupboards."

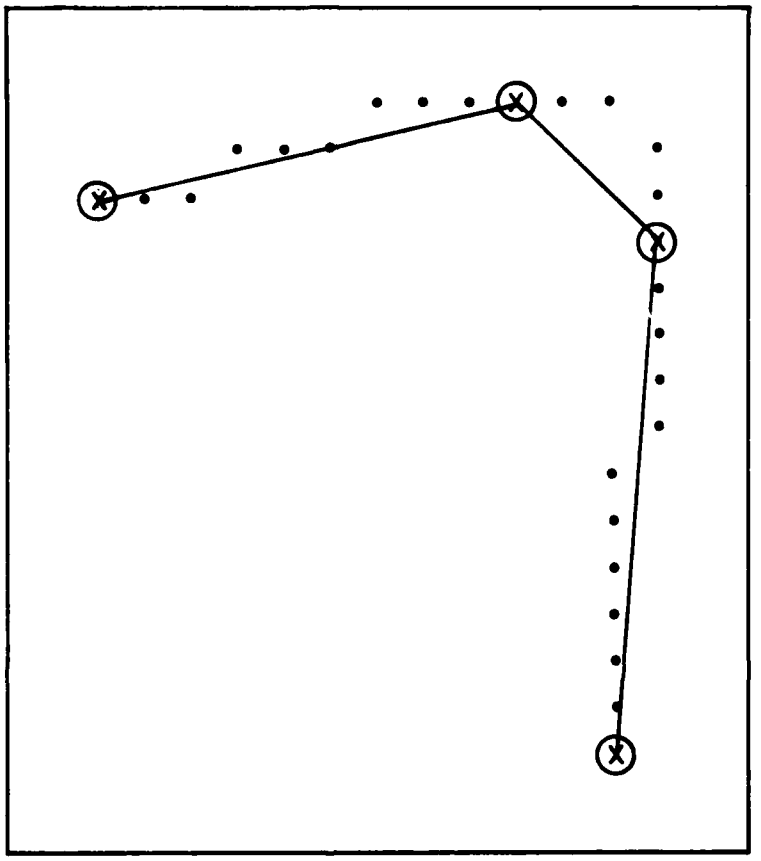


Figure 11. Rounding effect of earlier linear approximation algorithms

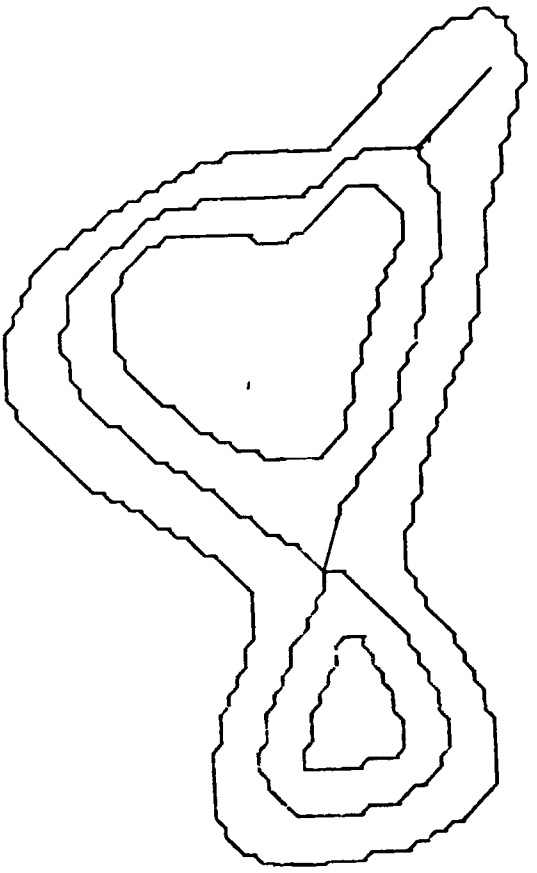


Figure 12. Spur artifact

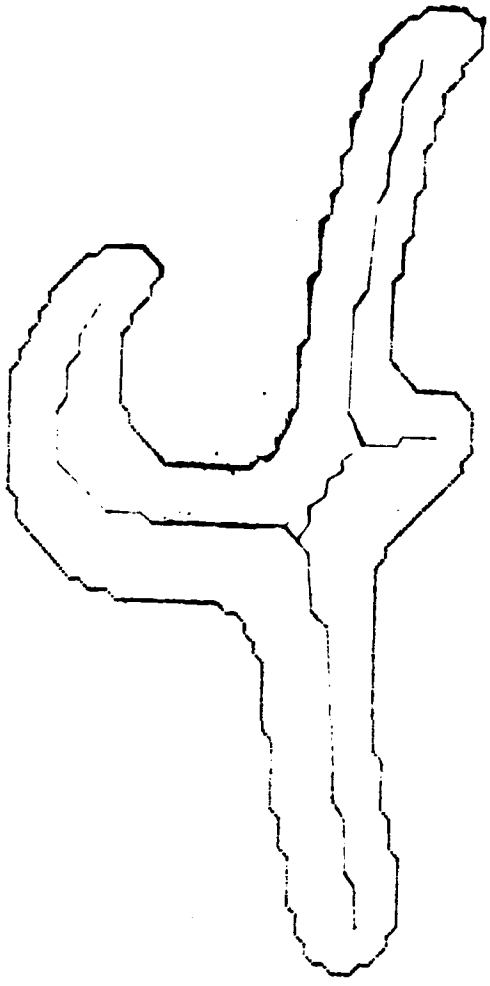


Figure 13. Improper representation of stroke crossing

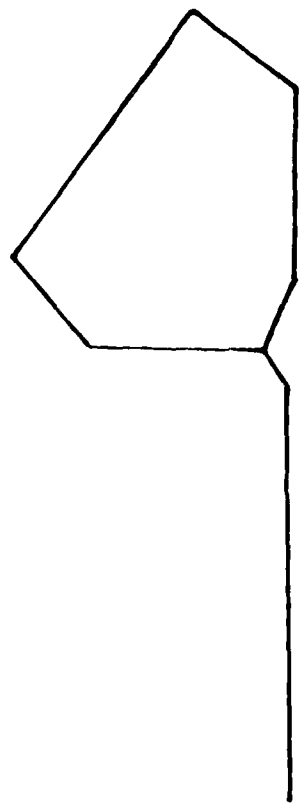


Figure 14. Displaced branch point

linear approximations have been generated. These two processes, however, do not create these artifacts; they are present in the input line image and are generated in the original process of skeletonizing the full raster image of the character.

Special algorithms have been developed to handle the skeleton artifacts of spurs and improper crosses within the HSR System itself. The analysis of the problem of displaced branch points has led to significant improvements in the thinning process. In particular, it has led to a complete re-evaluation of the use of the Medial Axis Transformation (MAT) (Blum, 1967) as a method of generating strokes (Brown, 1981; Cheng, 1983). Approximate versions of the MAT were used in earlier PAL work on CHITRA (Dasarathy, 1978; Lybanon and Gronmeyer, 1979; Lybanon and Kumar, 1979) and in the skeletonizing routines used in the RADC program called EXECIZ (Moritz, 1973). These routines were used to create input stick figures for HSR, Version 1.0. As a result of this analysis and experimentation, however, it was concluded that the MAT is not suitable for high-performance recognition systems based on the concept of "stroke representation" of handprinted characters (Brown, 1981a). These MAT-based thinning algorithms have been replaced as an input preprocessor for HSR, Version 2.0 by the new NORDA stroke generation software. These algorithms will be presented in the NORDA Technical Note 210.

A discussion of the spur and mid-point elimination algorithms is presented in the following two sections. Both of these algorithms benefit from the "continuous stroke" or linear approximation representation of the character; that is, if a spur or a mid-point segment is to be removed from the character description, this operation does not involve all the original sample points in the input image description of the character. At the linear approximation level, such adjustments only involve moving the terminators of the lineal elements that make up the connecting parts of this representation. This capability is an important by-product of the "linear segment" approach and makes such artifact removal a feasible process.

4.5 SPUR REMOVAL

Spurs have been a consistent problem for recognition algorithms that depend on a thin-linear representation of a symbol generated from a raster image. Spurs are often created during the "skeletoning" transformation of a raster image into a "thin-line," particularly those transformations based on the MAT (Duda and Hart, 1973). The problem results from the fact that a "spur segment" so generated appears just like any other segment which might be information carrying relative to the recognition logic.

Figure 15 is an example of raster images and the transformed stick figures which illustrates the problem of spurs. The raster image shown in Figure 15 is recognizable as the numeral "7." After the "thinning transformation," the stick image becomes a numeral "7" with a definite spur at the upper-right corner. To

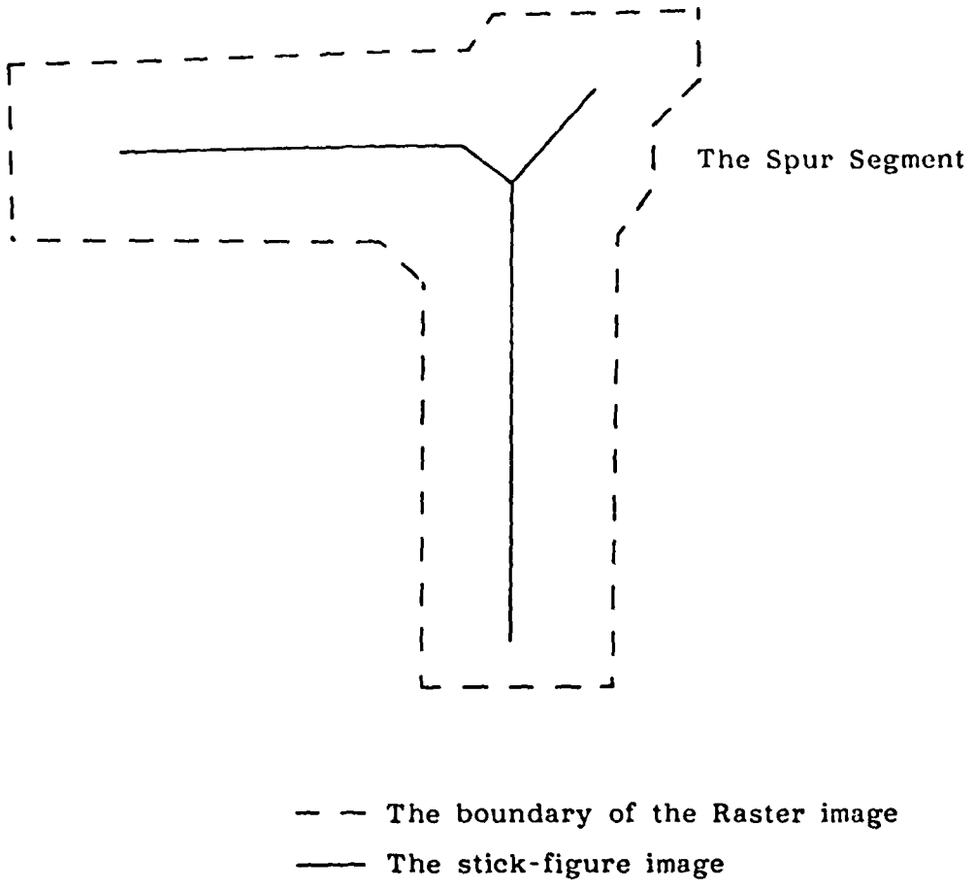


Figure 15. Spur generated by MAT-type thinning

illustrate the difficulty associated with this spur; however, assume that the raster region in the upper-right corner of the original character is extended gradually towards the right as shown in Figure 16. Under this circumstance the character becomes ambiguous and it is impossible to distinguish it as a numeral "7" or as an English letter "T." Thus, it is sometimes difficult to determine whether a segment is a spur or not (see also Blesser, et al., 1973).

Another example of a similar problem is shown in Figure 17. The stick image shown in this figure consists of three segments and is neither a conventional numeral "3" nor a conventional numeral "5." Without careful analysis and the use of a model of handprinting, this image would appear to have the characteristics of both. If segment 1 is identified as the spur and removed, then the other 2 segments appear to represent the numeral "5." On the other hand, if segment 2 is removed as the spur, then the remaining segments appear to represent the numeral "3." Therefore, the distinction between a segment and a spur in the stick-figure image after it has been "thinned" becomes ambiguous and a decision based simply on length is unreliable. Thus, one can see that the occurrence of spurs is often varied and unpredictable.

The examples given were generated by an approximate version of the MAT. Basically, these MAT-type algorithms attempt to thin the character by repeatedly stripping off the black points on the boundary of the raster image until only the "centerline" remains. This transformation technique inherently has the possibility of creating spurs (Duda and Hart, 1973). In other words, this technique basically fails to recover the original strokes from which the symbol was created (Brown, 1981b). The following discussions describe two specific techniques which can be used to minimize the effects of spurs when they are present in the stick figure before they are sent to the shape analyzer and the recognition logic.

In the data sets generated with the older MAT-type thinning used during the early HSR experiments, ten percent of the characters were observed to have one or more spurs. This percentage was obviously too high to be ignored during the HSR development. One solution to the spur problem is to employ the quality assurance module of the HSR System to classify "spurred images" as unknown characters on the grounds that the "conventional strokes" required for "good numerals" are not presented. In this way the accuracy of recognizing isolated characters can be maintained. This technique was the one employed in HSR, Version 1.0. However, when using this method, efficiency may suffer and sometimes fall below acceptable limits. The trade-off between such improved accuracy and reduced efficiencies depends on the extent of the spur problem in the particular data set being considered.

A second solution to the spur problem is to identify and remove the spur(s) from the stick-figure images. If successful, the images can then be recognized correctly after the spurs have been

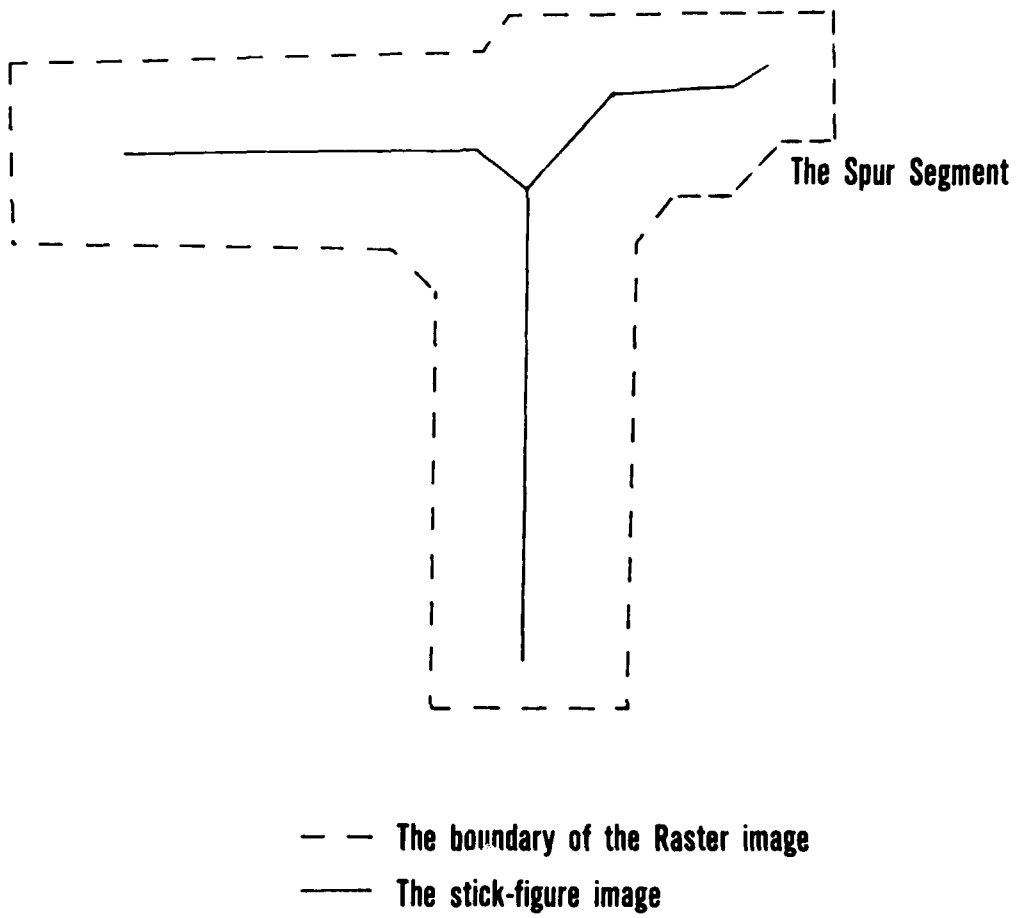


Figure 16. When is a spur not a spur? Compare to Figure 15 for "T" versus "7" ambiguity

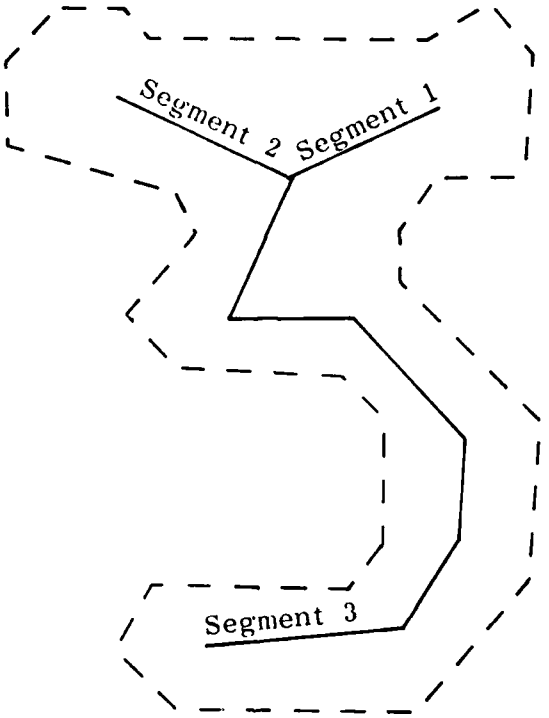


Figure 17. "3" versus "5" ambiguity in approximate image generated by MAT-type thinning

removed. Such a procedure should lead to an increase in the recognition efficiency. But the images can also be misrecognized if a segment is incorrectly identified as a spur. Therefore, a potential increase in substitution error exists when such a spur removal technique is employed; that is, a lower accuracy can result. The SPURS module currently used as an option in HSR, Version 2.0, was constructed not only to eliminate spurs but also to minimize the possibility of improper spur removal. Therefore, if a spur cannot be identified with a high degree of certainty, then the character is passed unmodified to the recognizer. If these characters did indeed contain spurs, they are likely to be rejected by the quality assurance module of the recognizer and no substitution errors will be generated.

A third solution to the spur problem involves a redesign of the fundamental algorithms for the reconstruction of strokes from raster images. As indicated in the preceding Sections, results of the NORDA investigations into new preprocessing procedures for this third solution will be presented in the NORDA Technical Note 210.

The development of global spur recognition algorithm to implement the second solution listed above is a very difficult problem as illustrated in Figures 18 and 19. Such a development is especially complicated if spur identification failures are taken into consideration. Thus, the module SPURS was designed to identify only certain specific and "insured spurs" under the guideline that a substitution error resulting from the spur removal operation must not occur. In order to develop an operational definition of a spur, a detailed analysis of the length, location, and characteristics of know, human-identified spurs was performed. As a result, the typical spurs which can occur for each numeral class were determined. Figure 18 depicts these regularly occurring spur types; the types illustrated account for approximately 95% of all spurs found in the FY-82 PAL database.

For numerals containing no enclosed regions, i.e., 1, 2, 3, open-top 4, 5, 7, a spur often occurs at the sharp turns. These spurs appear to be the result of the spreading of ink during the acceleration of the pen at such a corners in these numerals. For characters containing an enclosed region, i.e., 0, 4, 6, 8, and 9, a spur may occur as a result of "improper closing" of the region by the closing stroke.

With this background concerning where spurs occur, the following criteria were developed for the detection of spurs:

(1) A spur must have a branch point of type 3 and an end point as its terminators.
(2) A spur must be relatively short in terms of the size of the character. In addition, a spur must be shorter than the other two segments terminating at the same branch point of type 3.

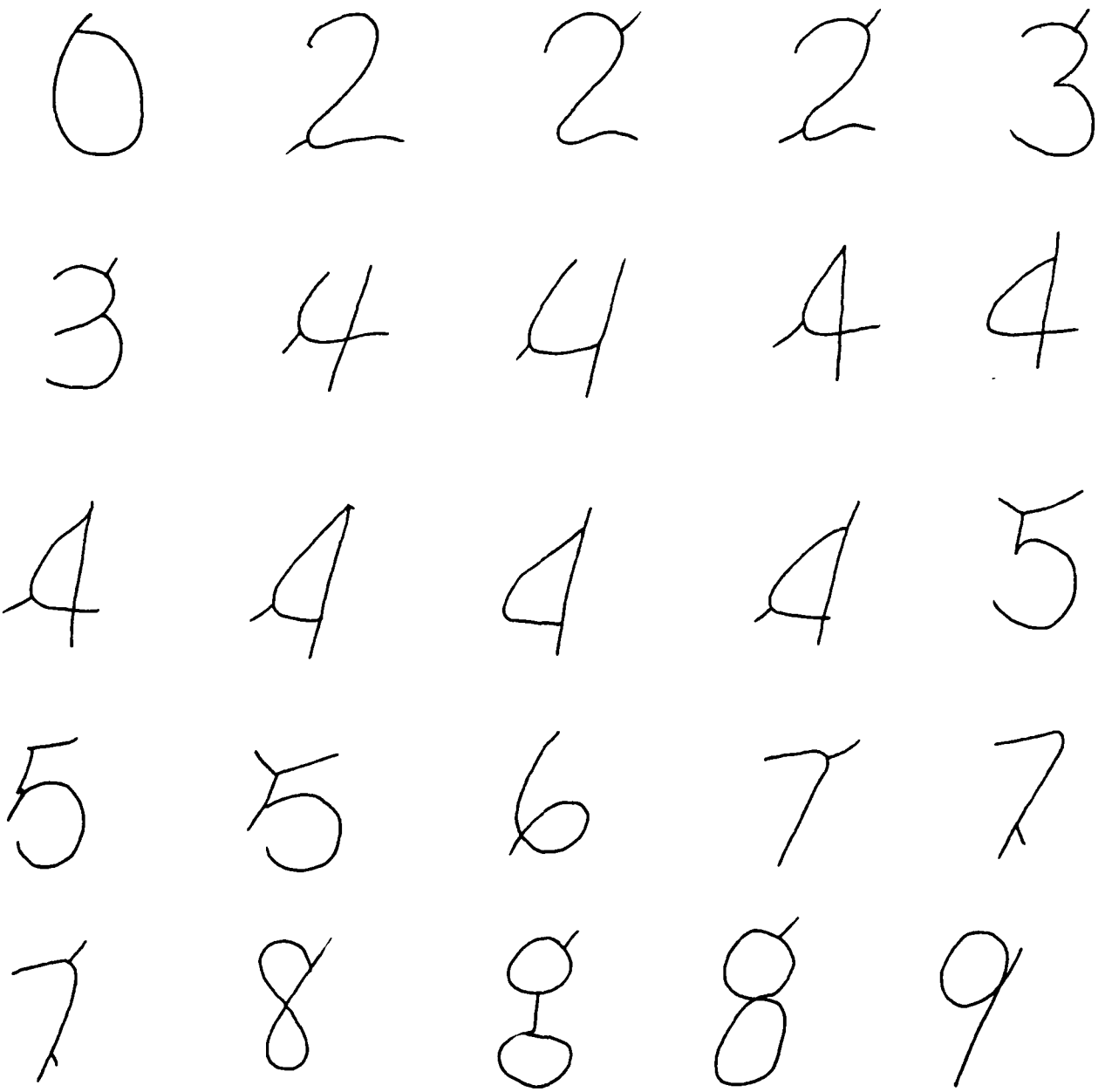


Figure 18. Types of spurs occurring on stick-figure images

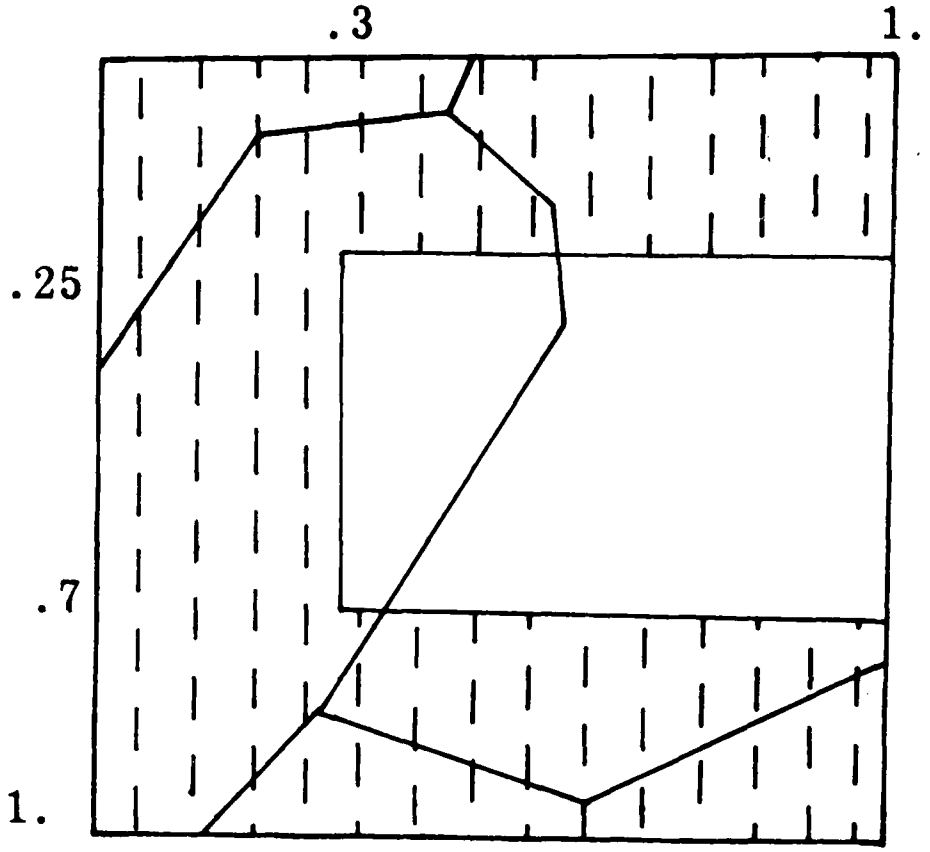


Figure 19. The feasible region for spurs

(3) A spur must appear within the "feasible region" of a character. The feasible region for a character is determined by placing a box around the character; the combination of the lower 25% and upper 70% of the height, and the left 30% of the width of this box is defined as the "feasible region"; this region is shown in Figure 11.

This feasible-region criteria is based on the extensive review of spurred characters just mentioned. Its purpose is to prevent the misidentification of the middle segment of numeral "3" and the right-hand segment of the numeral "4" as a spur. In fact, most all spurs observed are actually within the "feasible region," if the character is within 60° of being upright.

Table III in conjunction with Figure 20 presents an example of a minimum procedure for identifying two specific types of spurs that often occur on the numeral "2." The spurs S2 and S1 are removed after two passes from Step 1 through Step 10 in Table III. Such double spur removal illustrates the capability of this spur removal procedure iteratively to identify and remove multiple spurs from an input character. The procedure of identifying spurs on other numeric characters is similar to the one illustrated in Table III. The module SPURS consists of a collection of the subroutines needed to remove all the spurs depicted in Figure 18, which is a list of "empirical spurs."

The performance of SPURS was evaluated carefully through a number of experiments on the FY-81 PAL database. In the analysis of the experimental results, both the removal of segments which were not spurs as well as not removing human-identified spurs were considered as failures. The few failures identified by the experiment can be summarized as follows:

(1) Failure to identify other kinds of spurs not listed in Figure 18.
(2) Incorrectly identifying spurs on some numerals.
(3) Incorrectly identifying spurs in unknown characters.

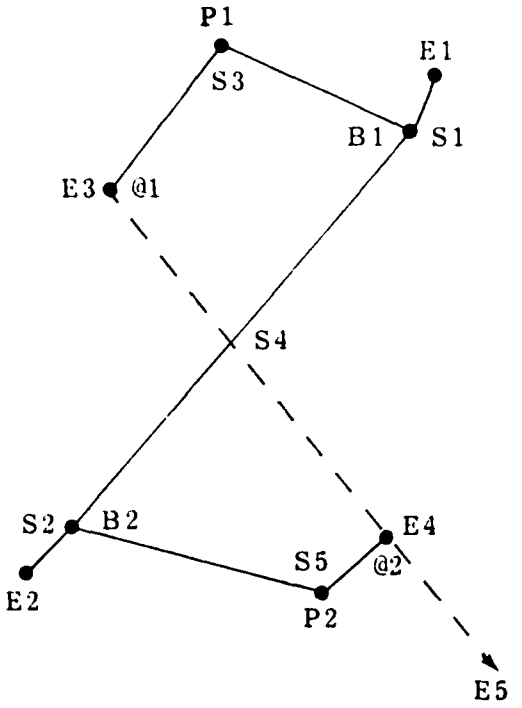
Fortunately, the failures were insignificant in comparison with the successes of SPURS. More important, the failures did not result in an increase of the substitution error of character recognition; characters containing such spur failures were rejected as unrecognizable. In conclusion, SPURS was designed specifically as a "spur recognizer" for the HSR development. Its accuracy in spur elimination considerably enhanced the efficiency of the character recognition process.

4.6. IMPROPER CROSS REMOVAL

There is another major type of artifact that results from medial axis thinning. This artifact can also be identified sometimes and then removed. It often occurs when one stroke crosses another stroke or itself; e.g., when an "x" or a "+" is handprinted. Furthermore, the width of ink spread by the pen at

TABLE III. Example Spur Removal

Step 1:	Sort all segments with one end point and one branch point of type 3 in ascending order of length and place them in "a potential spur list." In the example, this sorted list is S2, S1, S3, S5.
Step 2:	Try the shortest segment from this sorted array as the first spur candidate. The procedure will be terminated when the potential spur list is empty. In the example, the shortest segment is S2.
Step 3:	Calculate the ratio of the length of the spur candidate to the total character length. In the example, this ratio is approximately 0.05 or 5%.
Step 4:	Compare the ratio calculated by Step 3 against a predetermined threshold (10% assumed). If the ratio is less than the threshold, move to Step 5. Otherwise, increment the potential spur segment counter and go back to Step 2 to address the next potential spur in the list of sorted segments. In the example, one moves directly to Step 5.
Step 5:	Check if the spur candidate is within the feasible region. If so, move to Step 6; otherwise, increment potential spur segment counter and go to Step 2. In the example one moves to Step 6.
Step 6:	Draw a line between the two end points of the two longest segments in the sorted array to form a baseline from which angles can be determined. In the example, the end points for this axis are E3 and E4.
Step 7:	Calculate the two angles θ_1 and θ_2 measured from the extension of the baseline constructed in Step 6 to the approximate points P1 and P2, respectively. In the example, θ_1 is approximately 60° and θ_2 is approximately 150°.
Step 8:	Check to see that the two angles are consistent with a target numeral category. For example, θ_1 must be positive and θ_2 must be negative for numeral "2." If so, the character is probably the numeral "2," otherwise, the character is another numeral. Example: Move to Step 9.
Step 9:	Perform other target numeral assurance tests like in Step 8. If the tests succeed, move to Step 10, otherwise, move back to Step 2. Example: Move to Step 10.
Step 10:	Remove the black points making up the spur segment from the segment list structure and the segment description list.



Segments = S1, S2, S3, S4, S5

End Points = E1, E2, E3, E4

Branch Points of Type 3 = B1, B2

Approximate Points = B1, B2, E1, E2, E3, E4, P1, P2

Figure 20. Spur removal example; use in conjunction with Table III

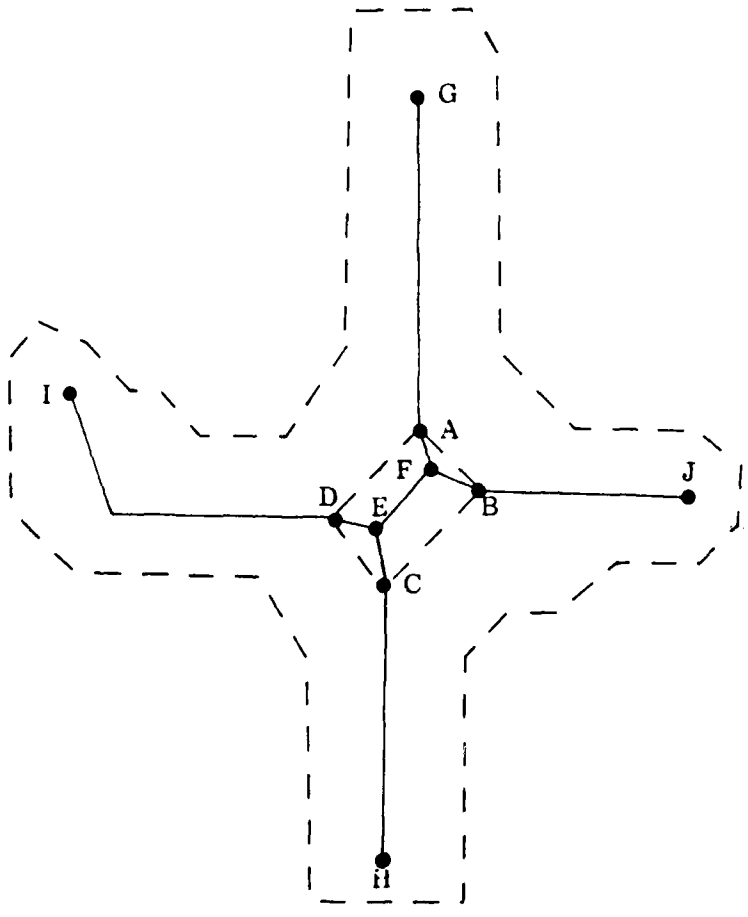
such intersections is usually wider than the average width along the main part of the stroke. When the preprocessor begins to thin a raster character which has such a "cross," the raster region surrounding the intersection point is gradually reduced to a region approximately outlined by four extreme points shown in Figure 21. Such raster regions should theoretically be in a shape of a "diamond-like square." If the length of one side of the "polygon" is out of proportion to the length of either neighboring side, however, the region will eventually be thinned to a short "artificial" segment connecting two branch points of type 3. The length of this short segment depends upon the angle which the overlapping strokes make with each other, the length of the "polygon," the area that the ink spreads at the intersection, the precision of image digitization, etc. More importantly, however, the topology of this representation (number of segments, branch points, connectivity) is incorrect.

The function of the "improper cross" processor, called MIDPT, is to find such artificial segments and reduce them to a single branch point of type 4. This newly generated representation for the two original crossing strokes by a branch point of type 4 eliminates the misrepresentation by "a short segment joining two improper branch points of type 3." First, MIDPT finds this type of "false segment" based on the following conditions:

(1) The segment must be terminated at both ends by branch points of type 3.
(2) The segment must be less than a predefined threshold; 10% of the total length of the character has been found adequate for the line weight used on smooth sheets.
(3) The segment must not enclose a region; for example, the top loop segment in a numeral "8."

After the "false segment" has been found, MIDPT calculates the midpoint of the middle short segment. This calculated midpoint is then used as a new terminator for each of the two pairs of linear segment elements which were originally attached to the two type 3 branch points. This newly constructed branch point of type 4 then replaces these two branch points of type 3 on each end of the "false segment." This modification in the linear approximation can easily be made by updating the segment tables and approximation point lists.

This improper cross phenomenon occurs most often on the numerals "4" and "8." Figure 22 presents two examples of the numeral "4"; and Figure 23 presents two examples of the numeral "8". As shown in these figures, the stick images are modified by MIDPT to re-establish the original stroke information. The elimination of the artificial segment occurring at the "intersection point" of the stroke(s) reduces the variety of topology which must be considered for the numerals "4" and "8," and brings the topology into agreement with the human-intended strokes.



A, B, C, D are the four extreme points

$\overline{EF}$ is the final branch 3 to branch 3 segment

$\overline{GA}$, $\overline{JB}$, $\overline{HC}$ and $\overline{ID}$ are the four segments connected with $\overline{EF}$ segment

Figure 21. Polygon source of improper crosses generated by MAT-type thinning

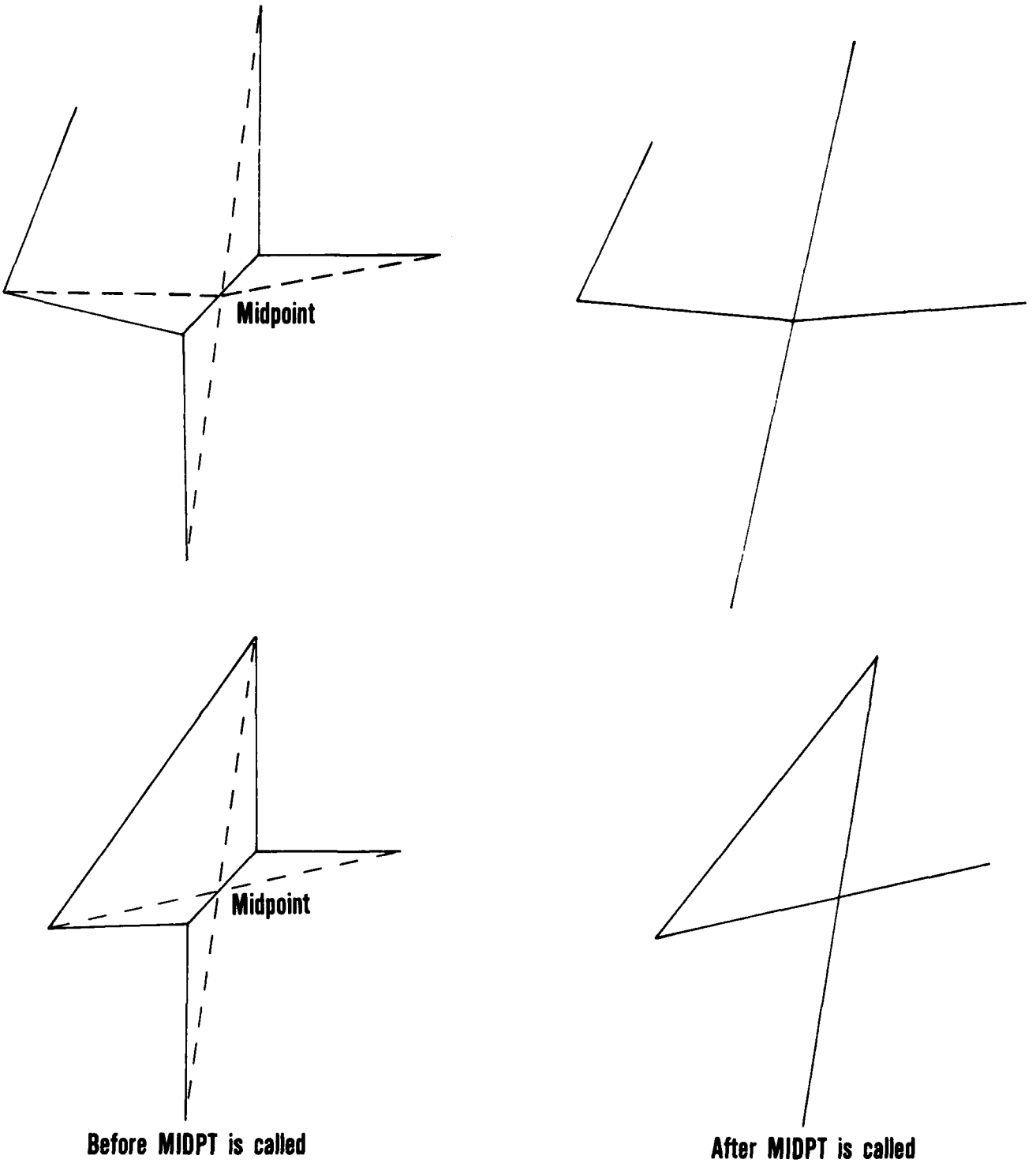


Figure 22. Removal of the improper crosses for the numeral "4"

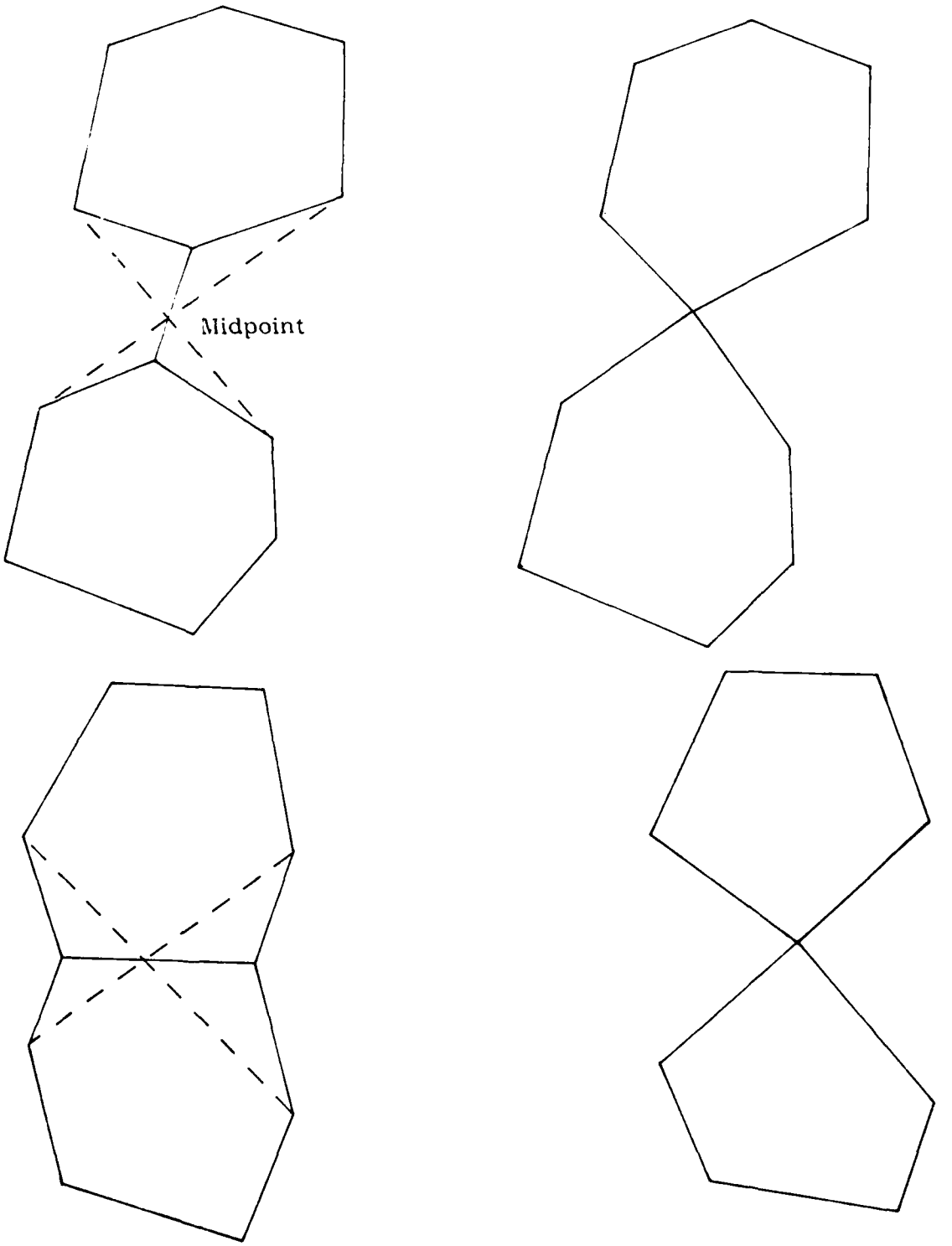


Figure 23. Removal of the improper crosses for the numeral "8"

4.7 GUIDELINES FOR FEATURE EXTRACTION AND RECOGNITION PROCESSING

As indicated in the introduction to this chapter, there are three key, interacting parts to the overall character recognition problem: (1) preprocessing; (2) feature extraction; and (3) recognition or decision logic. The next section considers curve analysis and measurement topics which are on the borderline between preprocessing and feature extraction. Section 4.9 is concerned with the feature extraction module which, in addition to interfacing with character image preparation and restructuring stage of the processing, also interacts strongly with the recognition module. As background for these discussions, this section outlines several general principles which have guided the PAL advanced development efforts for the last two major HSR functions: shape extraction and decision tree processing or recognition.

A three-module approach to recognition has been a key concept in the development of the HSR System: (1) pre-recognition, (2) potential numeral identification, and (3) final quality assurance (Brown and Gronmeyer, 1980). The "pre-recognition stage" is designed to separate "all possible non-meaningful images" from those that the recognizer has a chance of identifying with minimal misclassification. This pre-recognition stage is a difficult step to implement. This rejection channel must not be so stringent that it flags as unrecognizable those symbols that are basically acceptable to the recognition module; however, it must, to the extent possible, keep "trash" out of the recognizer subsystem that would increase processing time and substitution errors. This process is called pre-recognize in the following sense: the features in this module are specifically designed to recognize "trash"; that is, to identify the fact that a legitimate character is not present. Examples of features vector components that measure "trash characteristics" are the height (h) and width (w) of the object and their ratio (w/h), the density of points, a measurement of the number of holes or breaks, and the complexity and number of curve segments. Furthermore, since these "trash features" are specifically designed to perform this "garbage collection" function, not all of them are useful in actually classifying characters.

If an image is not rejected by the "trash filter," it passes to the symbol identification module. This part of the recognition process uses a feature selection/decision tree approach in which the most reliable and "rugged" features (relative to real-world scanner data) have been placed at the top of the tree.* One of the

* This structuring is in contrast to earlier work (e.g., Lybanon and Gronmeyer, 1978; Dasarathy and Kumar, 1978; Gronmeyer and Ruffin, 1978) where the best features relative to a theoretical class separation of ideal characters are calculated first. Furthermore, in the past, many recognition systems have been based on theoretical grounds and idealized input data. When such approaches are used for the present problem, an unacceptable drop in accuracy occurs (Brown, et al., 1979).

principal tasks in developing a recognition system is the definition of such features (Pavlidis, 1980). It is important to note that, in the long run, the determination of the suitability of a given feature concept is an empirical problem, particularly in view of the complex symbol environment being considered.

Extensive iterative experiments on feature definition and usage have been carried out at the PAL in parallel with the overall classifier design. It was found that many features used by earlier researchers are particularly sensitive to minor subtleties in sampling, preprocessing, and intra-class variation (Brown, and Gronmeyer, 1980; Gonzalez, 1980). The successful features emphasize global properties of the symbol being analyzed:

- Their calculation involves as many image points as possible.
- They organize the symbol information content in terms of the segments and strokes which compose it.
- The number of global segments, their relative orientation, and connectivity are rugged features, since they depend on the reasonably stable properties of end points, branch points, and enclosed regions.
- Individual segments and macrosegments are analyzed, in toto, for their general shape; e.g., curved, straight, spiral, etc.
- These features make extensive use of the concepts of rotation independence and the relative angles between various parts of the symbol.

In all the recognition modules, but most importantly in the identification module, a logic tree approach is employed which can operate in an iteractive manner with the feature extraction process. Organization of the recognition logic as a binary decision tree provides an effective mechanism for implementing feature selection and experimental evaluation. Such a structure can also be viewed as a recursive system with a feedback path between the classifier and the feature extractor.

After a symbol has been tentatively identified, more precise properties can be measured to determine if it is indeed the particular symbol in question. This function is performed by the post-recognition or quality assurance module. The distinction between identification and such verification is primarily one of emphasis. It is a convenient guiding principle, however, and makes possible the implementation of high confidence levels through the use of a final rejection channel for the symbols not meeting the quality assurance standards. By knowing that an image is tentatively a "5," for example, and that it has been separated from all other possible symbols (for example, that it is not a "2" or an "8"), more particular information or symbol specific features can be extracted which will lower the possibility of a misclassification or substitution error. (Note that the HSR System requirements place a higher cost penalty for substitutions than for lower throughput.) Thus, the task of this final recognition module is

not classification per se, but is one of establishing that the symbol meets a Sufficient Class Membership (SCM) criterion (Brown, et al., 1979).

A set of shape features for a character meets this SCM criterion if their specification "guarantees" that the pattern inside the subraster contains the necessary (sufficient set of) shape characteristics to justify, in terms of human recognizer judgments, the assignment to the character class in question. This concept is in contrast to a known-class separation criterion, which requires only that a set of features be based on the assumption of a predefined, finite universe of classes and does not easily handle the large, unknown number of classes represented by "trash."

The use of an SCM criterion effectively allows an expansion of the number of possible classes which the HSR System can handle lumping the "infinite trash universe" into a single rejection class. It is also a critical element in achieving the very low substitution rates required of the system.

4.8 CURVE ANALYSIS AND GEOMETRIC MEASUREMENT

The two major HSR functions discussed so far, namely, segment generation and artifact removal, have essentially prepared the input character image so that detailed analyses and measurements can be made about the shape of the curves out of which it is composed. This third basic function will be discussed in this Section. The curve analysis and measurement process is made up of two parts:

(1) Construction of "strokes" from several segments,
(2) Geometric measurements of these "strokes."

The definition of a segment in Section 4.2 indicates that any given segment in which the original input image has been decomposed or organized does not necessarily represent a handprinted stroke. This situation occurs because the segment generator only uses local connectivity (topology) in the segment list construction process. Therefore, if a character is drawn with more than one human-intended stroke, these strokes can not be properly represented in the segment list structures.

An example of this distinction between strokes and segments is shown in Figure 24 which depicts a numeral "4." This symbol is usually written with two strokes; i.e., one pen lift during the middle of the character printing. According to the segment description of this numeral, however, this character is classified as a three-segment image. The "continuous stroke" from "A" to "B" in Figure 24 is not represented in the segment list. This situation does not arise because of the slight indentation at the type 3 branch point; the more advanced NORDA thinning algorithms properly places this branch point essentially on a straight line

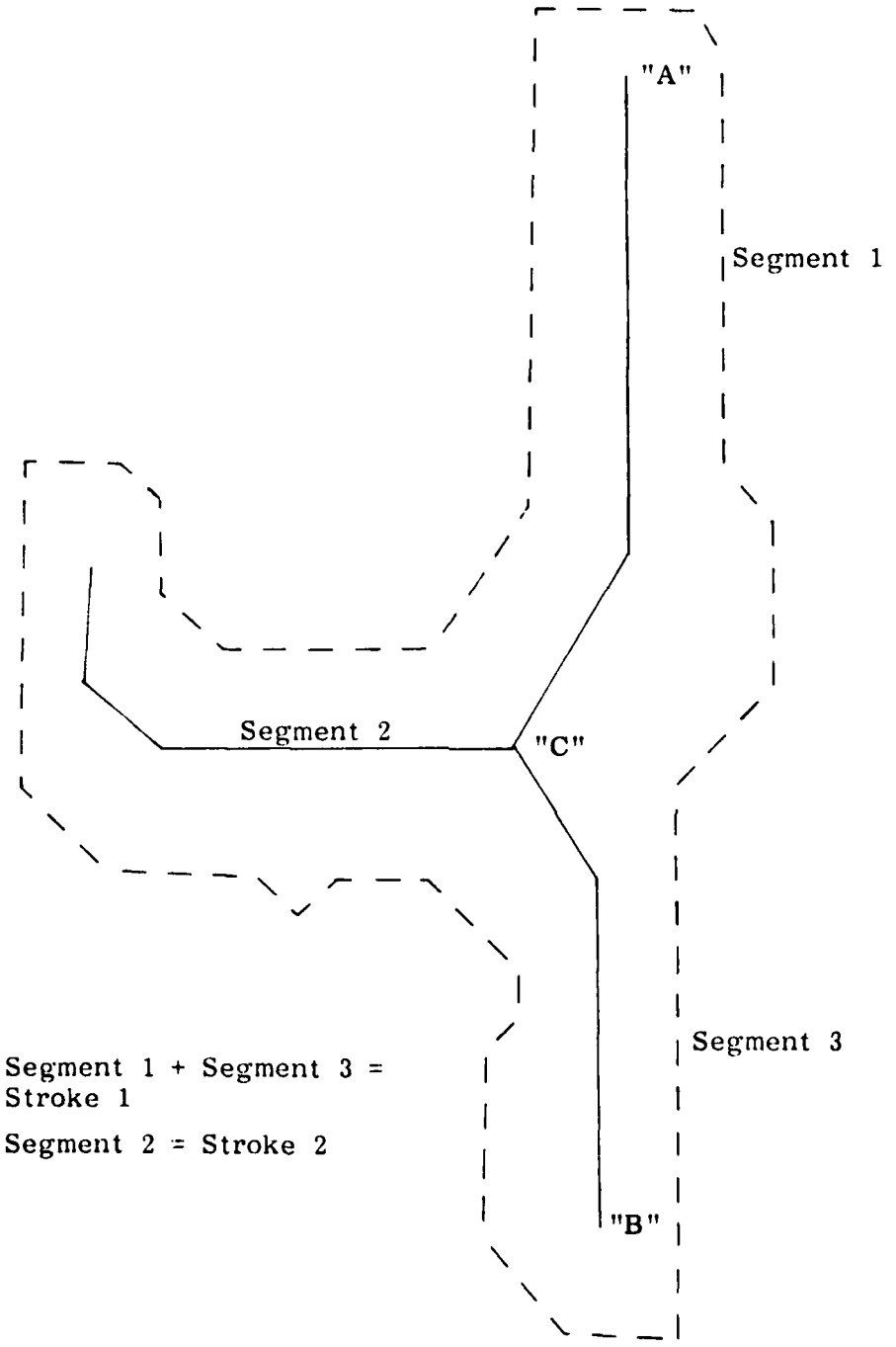


Figure 24. Strokes versus segments; example of a three-segment, two-stroke numeral "4"

joining points "A" and "B." The problem arises because of the fundamental difference in topologic descriptions required. There definitely is a topologic branch point at point "C"; however, one still needs to identify and describe the "connectedness" or continuity of segments 1 and 3 which make up stroke 1.

As one can see from this example, the differences between strokes and segments must be handled carefully in order to use the heuristic model of handprinting considered in Section 2.2. There are three approaches to this "stroke reconstruction problem":

(1) Use the segments just as they are generated and try to make allowance in the feature space measurements and the recognition logic to compensate for the fact that the "true strokes" are not being employed; that is, use the recognition logic "to reconstruct the strokes."
(2) Develop algorithms to continue the image transformation process begun by the first two major HSR functions so that the human-intended strokes are indeed generated. This approach would explicity define a new mathematical object called "stroke." These algorithms would then convert the input segments and associated segment lists into these new geometric objects and lists that describe them.
(3) Generate the set of all binary combinations of segments; this set will include the human-intended strokes along with other perhaps interesting combined segments; these binary combinations composed of two regular segments properly connected together are called macrosegments.

The earlier recognition program HSR, Version 1.0, employed the first approach listed above. It primarily extracted geometric features solely on the basis of each individual segment. An example of the compensation technique used is illustrated by the handling of the "4" in Figure 24:

- A measure of the straightness of segments 1 and 3 is generated by the curve analyzer; e.g., the length-to-width ratio of the box which encloses the segment,
- The angle between these segments is measured (assuming the lines joining the end points and branch point are straight),
- The recognition logic determines that segments 1 and 3 are "sufficiently straight" as measured by an appropriate threshold.
- The angle between the segments is tested; if it is near 180°, then the segments 1 and 3 have the essential properties required by the numeral "4"; that is, they form a "straight stroke."

This first approach worked remarkably well in HSR, Version 1.0, for many characters. However, this approach had one significant drawback. It did not allow the curve analyzer to capture certain interesting shape properties of multi-segments because all shape measurements were required to be made on individual,

"reasonably localized" segments. That is, the HSR, Version 1.0, curve analyzer could not directly measure the "shape properties of strokes." An example of this deficiency was illustrated in Figure 19 which depicts a numeral "2" generated by the earlier MAT-type thinning. The fundamental shape measurement for a "2" is a single stroke

- That crosses the line joining the end points of the stroke between the end points only once.
- One end point should be in the upper left part of the "picture frame"; the other end point should be in the lower right hand part of the "picture frame."
- The stroke first loops to the right and then to the left; these excursions of the loops should be approximately equal.

Although the above measurements are not the complete set used in the recognition of the numeral "2," they are part of the set of required properties; in particular, they indicate the "serpentine" shape of the "2." An examination of Figure 19 indicates that the curvature structure just described cannot easily be reconstructed from the three segment description of this character because the branch point is the lower left hand part of the picture frame breaks up the "continuous stroke" that is the "main body" of the "2."

Approach 1 to the stroke problem is reasonably straight forward; however, it leads to a higher rejection rate for characters than desirable. The next easiest approach is the macrosegment approach listed above as 3. This approach has been incorporated in HSR, Version 2.0. This technique uses a special processing module to construct a complete set of macrosegments which indeed includes all the human-intended strokes. The difference between it and the "true stroke" generation approach (2 above) is that this macrosegment processor does not have the sophisticated judgment capability to identify and remove the multi-segments which are not "true strokes."

This apparent disadvantage is offset by the fact that this macrosegment approach allows a certain kind of redundancy in the shape measurement process which can be exploited to obtain additional information: each macrosegment can be treated by the curve analyzer as a single segment; thus, a multi-segment input image is considered to be made up of several or more "synthetic single segment" images. An example of these multiple images is shown in Figure 25. Furthermore, there are characteristic shape properties in the "non-stroke" macrosegments which provide unique information for some characters. Even though these properties are abstract from the heuristic handprinting model point of view, they have contributed to the high performance of HSR, Version 2.0.

The macrosegment processing module takes every segment and combines it with another segment to form a macrosegment. This set of macrosegments, after duplications have been eliminated, forms

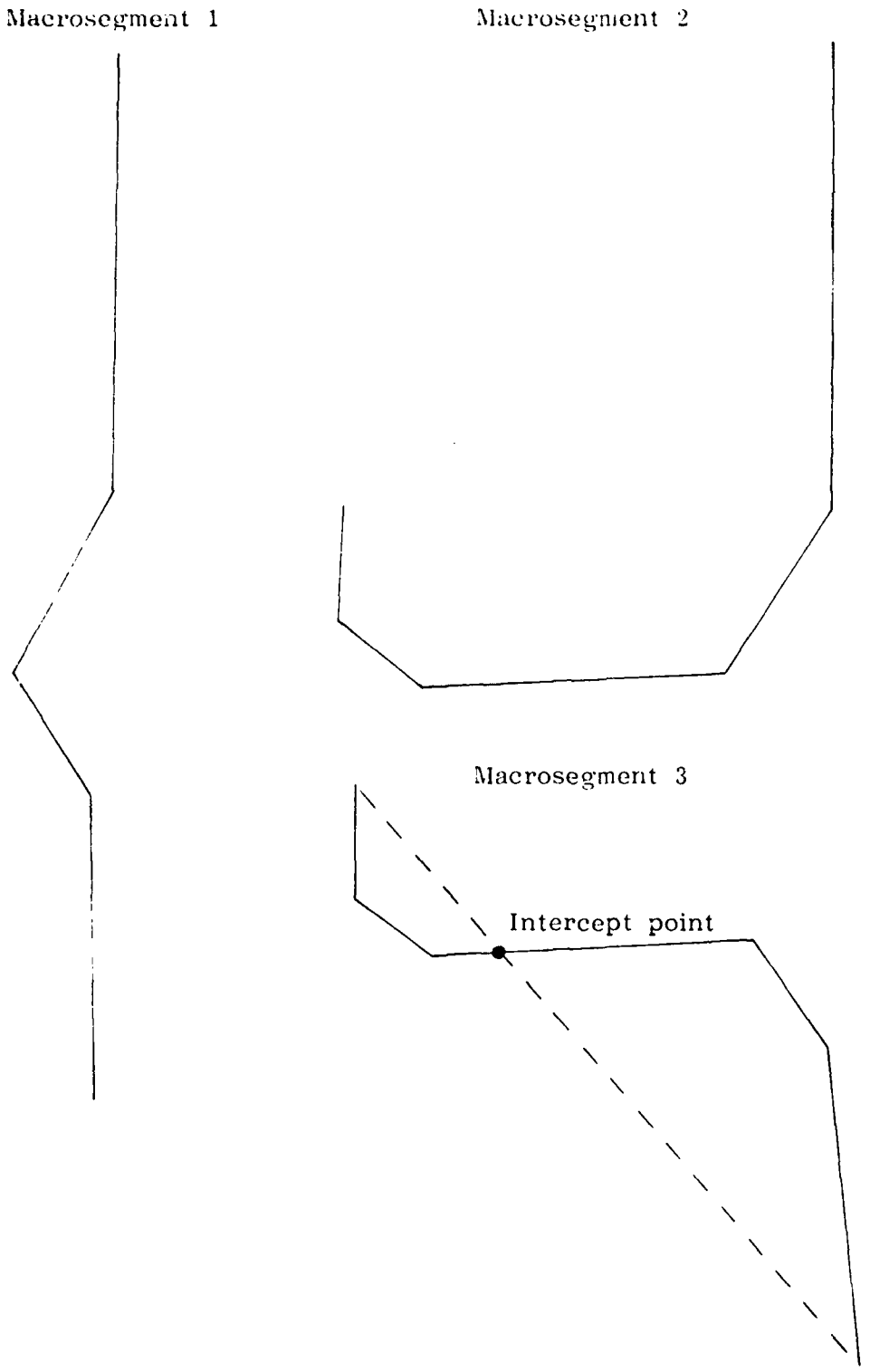


Figure 25. Three macrosegments based on the image shown in Figure 24

the basic description of the input character. Along with these macrosegments, the processor also generates a top-level macrosegment description list similar to the segment description list mentioned in Section 4.2. Thus, a one- or two-segment image is described by one macrosegment; a three-segment image is described by three macrosegments; and four-segment images have six macrosegments in their description.

The three macrosegment "images" depicted in Figure 25 are the basic description used for the numeral "4" shown in Figure 24. It is assumed that any character consisting of three such similar macrosegments could only be identified as the numeral "4." Furthermore, the intercept point shown on macrosegment 3 in Figure 25 is an example of an "abstract" feature occuring in most open-top 4's. This important intercept point feature is not obtained unless segment 2 has been combined with segment 3 into a macrosegment, which in this case is not a "true stroke."

The curve analyzer module processes macrosegments and segments to generate some very basic geometric information about input image. The inputs to this module are the concatenated lineal approximations of the (macro) segments. It was found during the development of HSR, Version 1.0, that the end points and branch points in a stick-figure image are reasonably stable. Furthermore, it was decided to construct a coordinate reference system for each segment based on these points which would be independent of the orientation of the image. This right-handed coordinate system used the line joining the end points of the segment as the X-axis or "stroke axis." This axis concept has been carried over into the macrosegment curve analyzer. The first operation of this module transforms the approximation points of the (macro)segment into this new coordinate system. In this system each (macro)segment is treated as a function (sometimes a multi-valued function). The (macro)segment is then traced from one end to the other. During this process the following information is recorded:

(1) The turning angle at each approximate point.
(2) The coordinates of the approximate points; that is, the perpendicular distance to the new axis and the parallel distance along the new axis.
(3) The coordinates of the local minima and maxima relative to the new axis.
(4) A "curvature code" for the whole (macro)segment which indicates the number and type of minima and maxima exhibited by the (macro)segment.

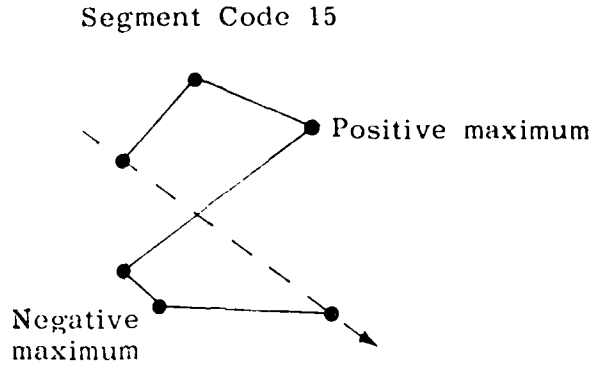
The current curvature code is constructed in the following way. Each minimum or maximum on the positive side of the axis is assigned a quantity of 10 and each minimum or maximum on the negative side is assigned a quantity of 5. These assigned values for a (macro)segment are summed up as the special curvature code to represent approximately what overall shape of the (macro)segment displays. A numeral "2," for example, would have a positive maximum followed by a negative maximum, and the curvature code should be

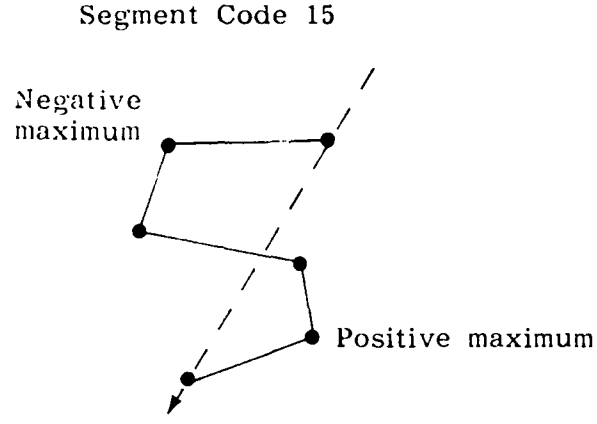
15. A numeral "5" would have a negative maximum followed by a positive maximum, and the curvature code should also be 15. In addition, the curvature code of a one-segment "3" should be 30 since it has one positive minimum between two positive maxima. Figure 26 depicts these three numeral examples.

Stroke drawn characters have another property which can be detected in the curve analysis processing. Empirical study of a wide range of handprinted characters indicates that a stroke is sometimes intentionally divided into "substrokes." The author prints this type of stroke in such a manner as to deliberately distinguish or mark two or more parts in the stroke even though he does not raise the writing instrument to cause this differentiation like he does between strokes. This phenomenon usually occurs at corners, e.g., in the numerals "4", "5" and "7." In these cases, the rapid acceleration of the pen is used to "break the stroke" into two parts (three parts for the numeral "5"). Furthermore, it appears that such points of stroke subdivision are definitely information carrying features of the character and are needed to distinguish some symbols. Consider, for example, the difference between the numeral "9" and a "triangle-top 4 which does not have a horizontal segment to the right of the main vertical stroke"; the left most triangle corner, in particular, is critical to the separation of these two numeral patterns.

When two or more parts of a stroke can be distinguished, they are called microsegments; this name is used since the "subdivision process" is the opposite of the "joining process" of macrosegments. Finding a general solution for the detection of microsegments is a very difficult problem. This situation is true, in part, because humans can employ subtle means for "marking or dividing" the two "micro" parts of the stroke. The current technique relies on the fact that a microsegment separation point often occurs at a maximum or minimum since this point is the location where a significant change in the direction of a stroke may take place. This "break-point" condition does not hold, however, for smooth arcs; examination of the turning angles near the microsegment break point (i.e., the examination of curvature in the mathematical sense) can be used to distinguish these two situations. Alternatively, one could generalize the concept of microsegment to include the subdivision of such slowly turning curves. For example, the right gentle loop or "bay" in the numeral "5" could be considered as made up of two parts: the outbound portion and the returning portion.

As a last example, consider the use of microsegmentation when the relationship between two similar features is calculated. For example, an input character with a segment code of 10 is separated into two microsegments at the maximum point as shown in Figure 27. In order to determine whether this character is an acceptable "7" or not, the length ratio of microsegment 1 and microsegment 2 is calculated as one of the important features. Although the threshold for this ratio can vary quite widely, it is at least reasonable to say that the length of microsegment 2 must be shorter than





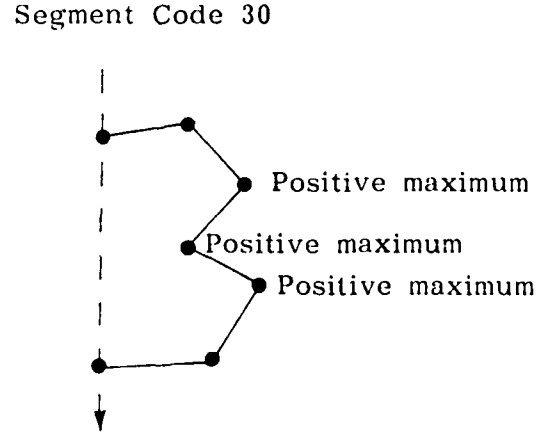


Figure 26. Segment code examples

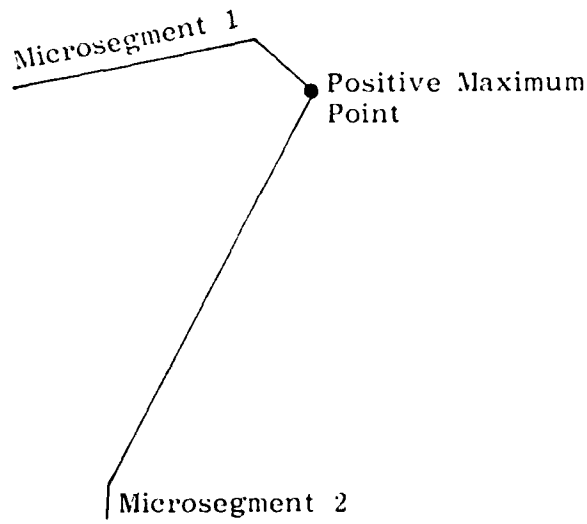


Figure 27. Two-microsegment numeral "7"

the length of microsegment 1 for the character is to be classified as the numeral "7." Using the microsegment technique to extract needed features makes the shape subroutines more complicated, but the features calculated through this technique have been found essential to high performance recognition.

The information generated in this curve analysis and measurement module is passed to the shape feature extraction module discussed in Section 4.9.

4.9 SHAPE INFORMATION AND FEATURE EXTRACTION

The feature extraction modules are designed to process the geometric information from the curve analysis and measurement module and the topologic information generated by the segment construction module. The shape features generated by these modules will be used as critical or defining properties necessary for the symbol identification. As indicated in Section 4.7, they should be "rugged" and to the extent possible, be independent of size, orientation, and authorship variation. These quantitative features are based on regular segments or macrosegments and must provide sufficient, i.e., adequate, descriptions of the input character so that the computer (or a human without seeing the character) can reliably identify the symbol by means of examining the values of these features. As indicated in the section 4.7, such feature definition or selection is a difficult area of research and is the key to high-performance, unconstrained handprint recognition.

The processing functions preceding this shape feature extraction stage have been independent of what symbols are to be recognized. The only assumptions about the symbols made up to this point are that (1) they are printed with a "thin-line" writing instrument, (2) the human-intended stroke model is appropriate, and (3) an individual symbol is distinguishable on the basis of its "measured" shape characteristics (in contrast, for example, to a context identification scheme such as used by humans to "fill in" missing letters in a word which has already been recognized by the meaning of the sentence).

To build a library of shape feature extraction algorithms for unconstrained handprinted symbols, however, one needs to consider the target character set. Indeed, in the case of numerals for example, the target set is larger than 10 elements since there are several varieties of patterns for some of the numbers: consider open-top 4's and triangle-top 4's, "regular" 7's and European 7's, 1's with and without pedestals, etc. Furthermore, whether a variation of a given character should be put in a separate "pattern class" partly depends on what shape measurements are being used. Thus, as stated earlier, the selection of shape measurements, the partitioning of the feature space, and the final recognition scheme are very much interconnected.

A wide range of different "types" of numeral symbols have been collected in the PAL database. These have been examined using

the heuristic model of handprinting discussed in Section 2.2, in order to find patterns or properties that would organize them into manageable categories or classes. If humans are asked to verbalize the questions they use to determine the labels for symbols, one can get an idea of the important factors, both perceived and actual, that are involved in human character recognition. These factors or criteria are part of the "code deciphering mode" discussed in Section 2.2; see also Blesser, et al., 1976 and Seun, et al., 1980. In fact, one of the approaches used in early stages of feature extraction studies was modeled after the game of "twenty questions." In this thought experiment, people are told that the investigator is "thinking of a symbol" or actually looking at one which he does not show them. The person is then requested to ask a minimum number of questions to determine what symbol the investigator is considering. It should be remembered that one of the main problems in the definition of shape features is to determine style-invariant, information-carrying properties of the symbols; this thought experiment allows one to concentrate on this communication channel.

This query technique combined with an extensive visual study of the symbols in the database and a review of the literature on shape analysis produced a wide range of suggestions for feature measurements. Algorithms to extract many of these properties have been developed at the PAL and histograms or co-histograms of these properties have been constructed to examine their stability and variability. In some cases, these shape measures have been used in recognition experiments to test their ability to partition the feature space. These investigations have laid the ground work for the current shape feature definitions. The heuristic methodology just described has been an iterative process and is still underway as new symbols are being added to the database.

On the basis of this early work, it was found that simple topology can be used as a property or criteria (question) in sorting out the symbols. Furthermore, these topology characteristics have proved to be relatively stable. For example, consider the numeral "8." Can a symbol whose stroke representation does not exhibit two enclosed regions be considered as a candidate "8"? The answer is basically no. Exceptions do exist, of course; consider the "8" in which the "closing part of the last stroke" does not cause the top part of the "8" to have an enclosed region. On the basis of the target numeral set concept just mentioned, open-top (incompletely enclosed region) "8's" and two-enclosed-region "8's" would be in separate pattern categories based on topology considerations.

The targeted numeral patterns used in the design of HSR, Version 2.0, are shown in Figures 28 and 29. These numeral patterns have been grouped by their simple topology measures: the number of segments (SEG) and the number of enclosed regions (ER). Symbols can exhibit a fairly wide range of shape variations within each of these categories and still be successfully recognized. Furthermore, the empirical study of the PAL database, both digital images

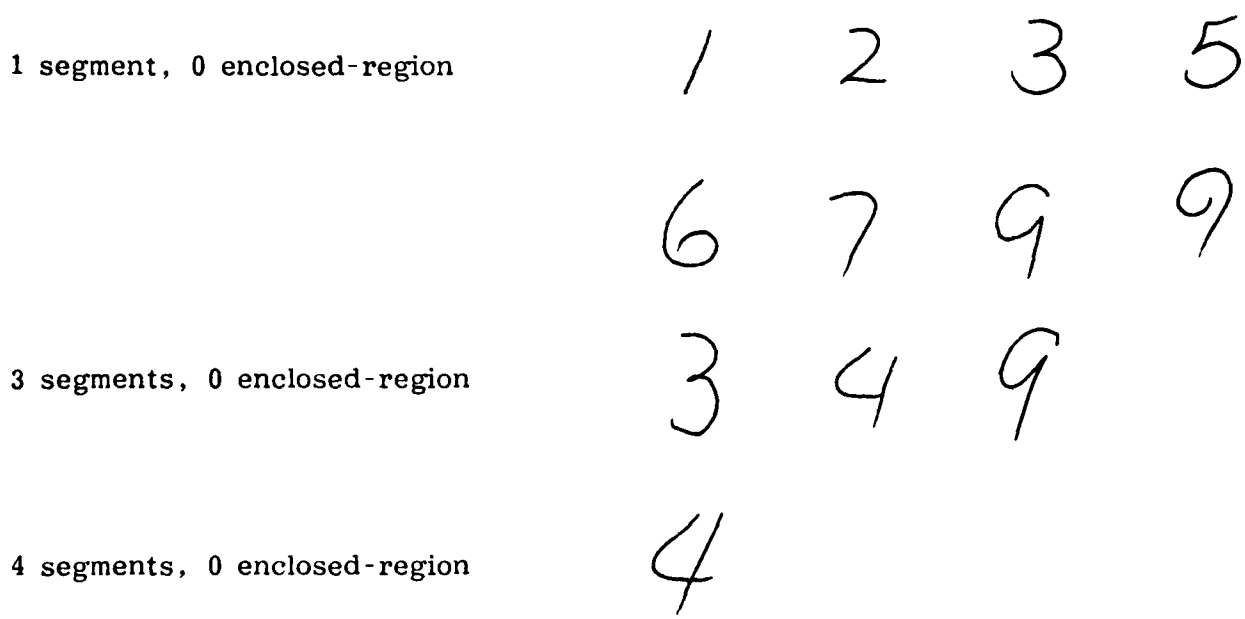


Figure 28. Numeral pattern categories with zero enclosed regions

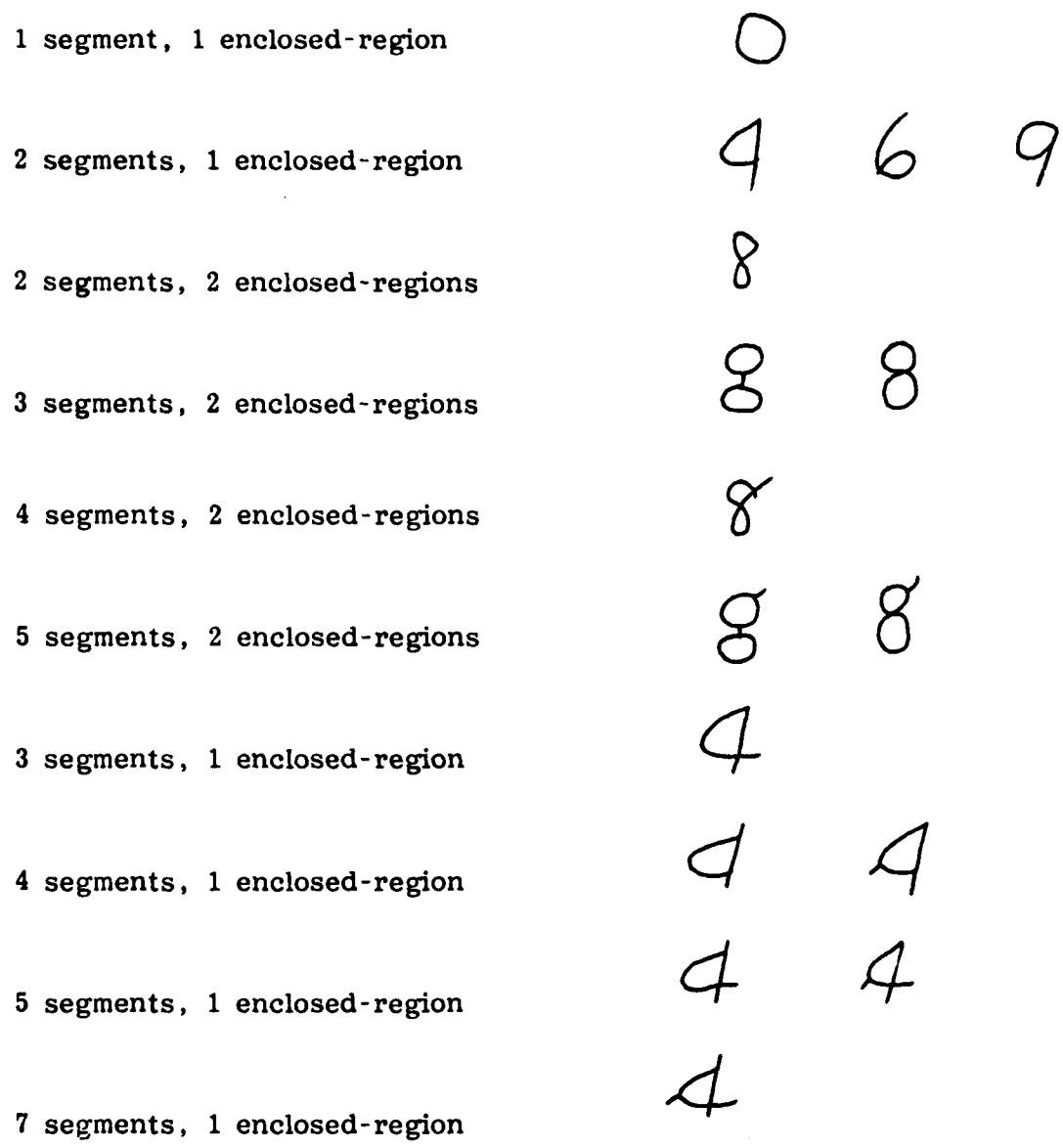


Figure 29. Numeral pattern categories with one or more enclosed regions

and M/C graphic documents, indicates that almost all handprinted numerals for these M/C products fall within one of these pattern categories.

Those pattern categories in Figure 29 with enclosed-region "4's" containing spurs and enclosed-region "8s" containing a segment joining two branch points of type 3 are not conventional or "human-intended" patterns for the numerals "4" and "8." However, the occasional failure of the artifact removal algorithms can result in these patterns. Because these patterns have such distinctive geometric features, they have been included in the HSR, Version 2.0, target symbol set and are treated as acceptable or recognizable numeral patterns by the feature extractor and recognition logic.

As a further example of the target categories, consider the numeral patterns shown in Figure 30. One pattern is a style variation for the number "2"; the other is a rather rare form of the number "3." Both patterns contain a small loop and are written by some people. Regardless of their popularity, however, these patterns have not been seen on the smooth sheets used in the HSR experiment.* Therefore, at present, the recognition program HSR, Version 2.0, does not consider these numeral patterns in its targeted character set. Because of the flexibility in how features and recognition logic are implemented in the HSR System, one could easily modify the software to recognize these patterns as well as others.

The numeral pattern categories listed in Figure 28 and 29 can be used to reject "trash," non-numerals, or even numerals that have been improperly preprocessed. As an example, consider the center-line stick-figures in Figure 31. These "stroke drawings" are shown in the context of the image from which they were "improperly derived." However, it would be incorrect to expand the feature space and recognition logic to try to incorporate these

*Two reasons appear to explain their absences. (1) All the characters on a smooth sheet are required to be quite small. This restriction usually results in the smaller enclosed regions in a character being filled-in by ink. In general, the small loop or enclosed region in this style of numeral "2" or "3" is much smaller than in numeral "4","6", "8", or "9." Therefore, the enclosed region in such numeral "2" or "3" is usually filled-in. (2) When a person writes a numeral for which the enclosed region must be present as an information carrying feature, he pays careful attention to these forms; he pays less attention to preventing an enclosed region in numeral "2" or "3" because he unconsciously knows that a filled-in "2" or "3" is still a recognizable numeral, i.e., the small loops appear not to be information carrying features. This "careful attention phenomenon" also occurs with numeral"6"; people can usually recognize this "distorted" character because of the surrounding context and the fact that the width of the filled-in region is twice as large as the average pen width.

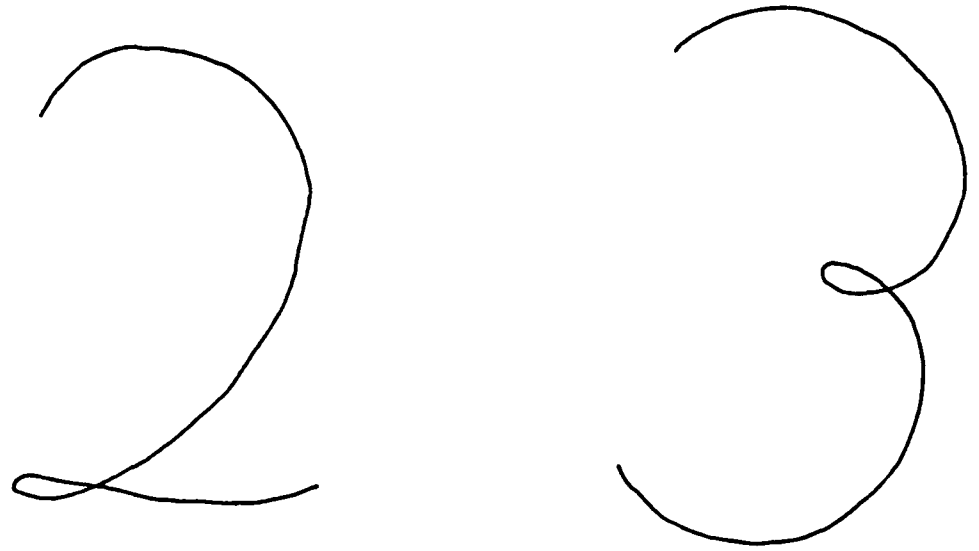


Figure 30. Patterns of the numerals "2" and "3" not found in the PAL database

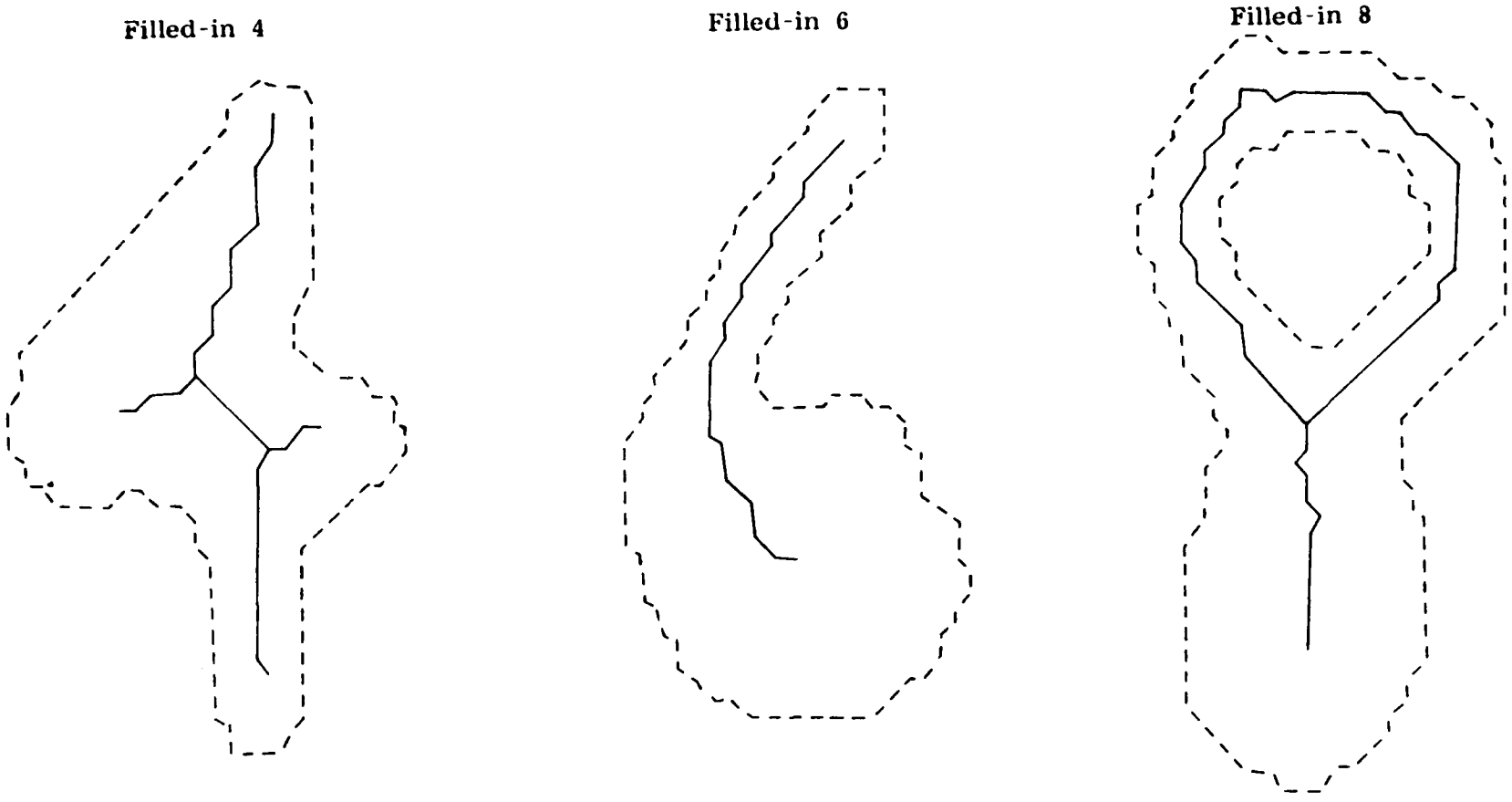
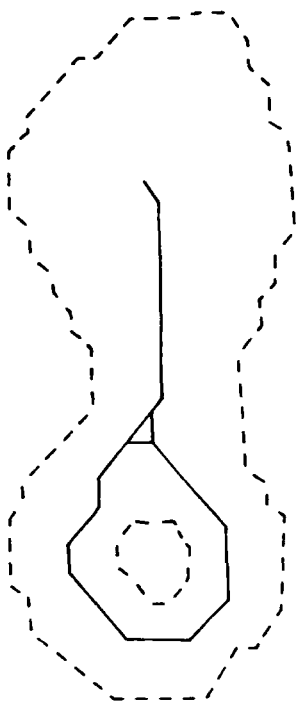
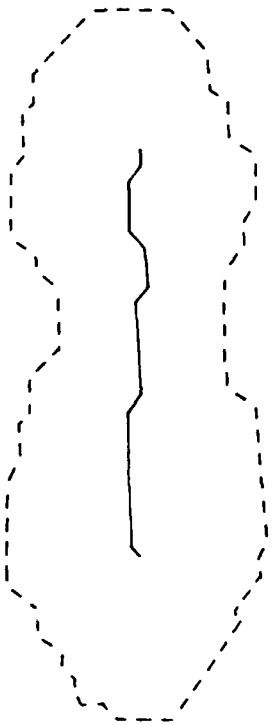


Figure 31. Filled-in images overlaid with their stick-figure images

Filled-in 8



Filled-in 8



Filled-in 9

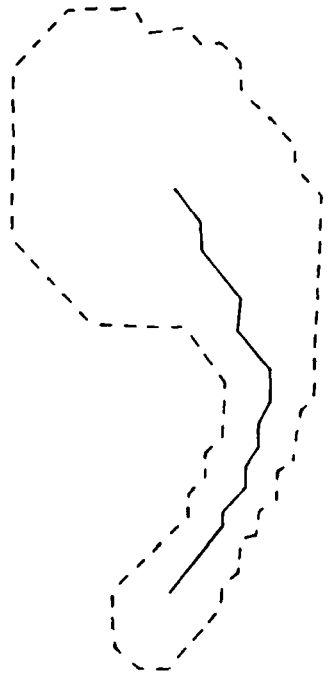


Figure 31, continued

types of filled-in problem numerals and still produce high performance recognition.

The feature extraction modules perform two basic functions: (1) they calculate additional shape information based on the relative geometry presented to it in the segments, microsegments, and macrosegments, and (2) they construct the actual feature vector components used by the decision (recognition) logic. In generating the final shape properties sent to the decision logic, the HSR System, Version 2.0, employs the significant concept of pattern-oriented feature extraction. It is clear that some properties of numerals are universal in nature in that they apply to, or can be calculated for, all character images. Examples of this type of feature are the topologic measurements: the number of endpoints, branch points, segments, enclosed regions, etc. (note that the Euler formula holds for these Euler properties). Similarly, a straightness measure can be made for all segments. To restrict the types of shape measurements to those that can only be applied to all images, however, would severely restrict the HSR System. Examples of non-universal features that are important include the following measurements: circularity for enclosed regions, angles between segments intersecting at a branch point, the ratio of area of an enclosed region to the "area of the whole image," relative positions of branch points, relative lengths of segments or microsegments, and ratios or relative positions of extrema. All of these examples can only be calculated for characters which possess certain shape properties.

The pattern-oriented feature extraction concept takes this situation into account; it orders the shape feature processing so that one calculation can build upon the results of another. Thus, for example, knowledge of the curvature codes generated by the curve analyzer can be used to separate two main subsets of feature calculations for three-segment images. One subset concentrates on the numeral pattern category "3"; the other subset concentrates on the numeral patterns for "4" and "9." The three-segment, zero-enclosed region subroutine first checks the three macrosegment curvature codes from a three-segment image to determine which subset of feature calculations should be used in further processing. As shown in Figure 28, it is obvious that the three macrosemgments description of a "3-like image" is quite different from the macrosegment description of a "4-like" or a "9-like image." A "3-like image" should have the three macrosegment curvature codes of 10, 10, and 30, but a "4-like" or a "9-like image" should have at least one macrosegment curvatrure code of 15 as shown in Figure 25.

Pattern-oriented feature extraction allows the subroutines of the feature extraction modules to distinguish and measure the fundamental stroke structure of a character. Based on this capability, each subroutine can generate image specific shape features, and therefore provides an optimum description of a character.

It is not feasible in this top-level review of the HSR System to discuss the details of all the shape feature measures. A few more examples will be given, however, to help illustrate the type of shape processing that is involved. The relative positions of endpoints and branch points are generated; these properties are used, for example, in distinguishing the numerals "6" and "9." The number, types, and position of the crossings which a segment or macrosegment makes with the axis joining its endpoints (stroke axis) are also recorded; in the single segment case, these features are used in analyzing "2," "5," and "7." The "beginning and ending angles" of a segment relative to the stroke axis are also useful properties; this measure, along with the straightness and intersection angle of the microsegments of an images, can be used to determine the numeral "7."

4.10 DECISION LOGIC AND RECOGNITION PROCESSING

This section presents an overview of the fifth and final major HSR processing function: decision tree processing or recognition. As indicated in Section 2.1, the critical problem in unconstrained handprinted character recognition is the definition and selection of the important information-carrying shape features. The success of these measurements is determined by their proper partitioning of the feature space into a non-overlapping regions; that is, whether they uniquely describe all characters in the target pattern categories. However, this partitioning or separation of characters, also, depends in an important way on the decision mechanism used. A binary decision tree has been selected to implement the heuristic model of handprint recognition discussed in Section 2.2. Such binary logic trees are similar to linear classifiers. They have two important additional advantages that are are discussed in the next paragraphs: the recognition decisions are allowed to interact with the feature extraction process; decision trees provide a good insight mechanism as how the classification process is operating and allow easy manipulation and experimentation of both features and decisions.

Currently, the integration of the feature extraction and decision logic as indicated in Section 4.7 has been only partially implemented. The reasons for this are partly historical, partly for research ease, and partly related to logistics and software development. During the development of HSR, Version 1.0, it was convenient to calculate most of the features before entering the logic decision mechanism; for example, this procedure allowed histogramming of feature values. There is nothing inherent in the overall HSR design, however, which requires this separation. In fact, as the symbol processing has become more complex during the development of HSR, Version 2.0, the interaction between these two major aspects of the recognition system has become more inevitable. Thus, as mentioned above, the feature extractor needs the information (results) the decision mechanism in order to know which feature to calculate next, since all features are not meaningful for all symbols. Conversely, not all features are needed in the identification and verification of any given symbol.

A powerful language system has been developed to implement this feature measurement and decision logic integration; it is called the PAL TREE for Pattern Analysis Language Tree. This collection of FORTRAN subroutines is an interpreter specifically designed to make it easy to develop and modify binary decision tree structures concerning arbitrary feature systems. A detailed discussion of PAL TREE involves the concepts of virtual processors, language structures, code syntax, etc., and is beyond the scope of this report. In general terms, each program line in the interpreter describes an "operation" or "decision" and automatically points to the next program line depending on the outcome of the process. The top-level flow chart of a pattern analysis scheme can easily be placed in a one-to-one correspondence with the PAL TREE program lines. What "operations" or "decisions" are executed is contained in the "micro code" subroutine library of the PAL TREE System and can be tailored to meet the requirements for a specific symbol problem.

With the use of the PAL TREE language system, the researcher and designer can concentrate his attention on algorithms, features, decision structures, etc., quickly convert his ideas into executable form, and carry out recognition experiments to gather statistics concerning the effectiveness of these concepts. As noted, the development of a very high performance recognition system is strongly dependent on the empirical/statistical nature of the input data stream. PAL TREE allows such studies to be easily made many times, with variations in the logic/feature structure over thousands of characters in the database.

The intention of any given PAL TREE program is to provide recognition logic with enough "expert knowledge" to identify an input character as a specific numeral or as unrecognizable. This knowledge includes the criteria for all targeted pattern categories and the decision flow that can be followed to reach the final identification for a character. The HSR, Version 2.0, PAL TREE initially tests topologic features (e.g., number of segments, number of enclosed regions, etc.). The results of these tests branch the process into various pattern examiners or expert questioners. Each target pattern examiner begins by checking the necessary features which the input character should possess. The examiner then checks the less critical features of the character. Although the border between the less critical and necessary features is sometimes difficult to draw, during the design of the PAL TREE for HSR, Version 2.0, emphasis was placed on the importance of the features in terms of a targeted pattern. A character is rejected as unrecognizable by a specific examiner only after the failure of one necessary condition or after multiple failures of less critical features. A character rejected under these circumstances is almost always completely outside of all the target pattern categories and is usually very badly distorted.

Figure 32 is a top-level flow chart showing how the HSR, Version 2.0, PAL TREE branches to various pattern examiners. Each

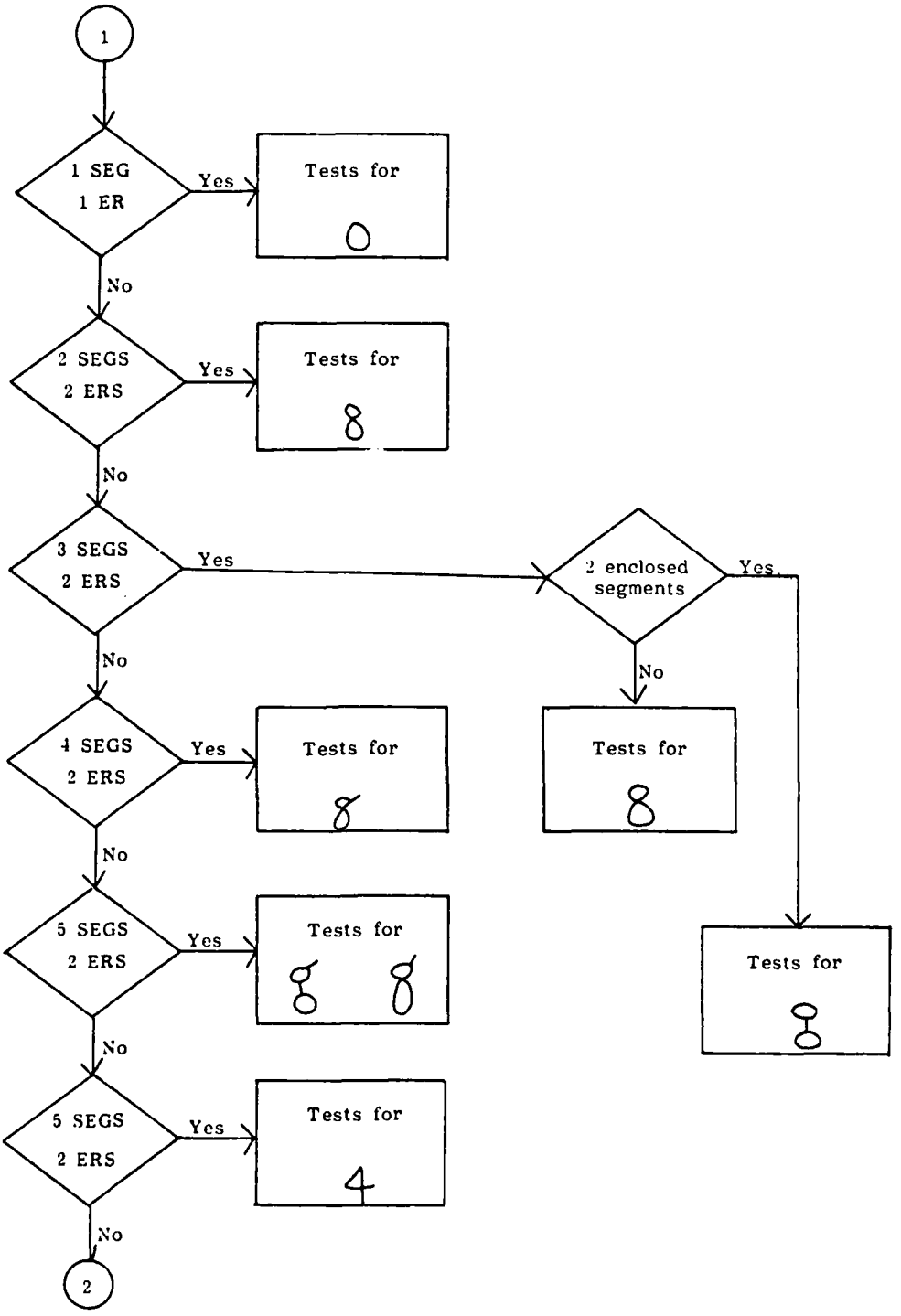


Figure 32. HSR, Version 2.0, PAL TREE (top-level structure)

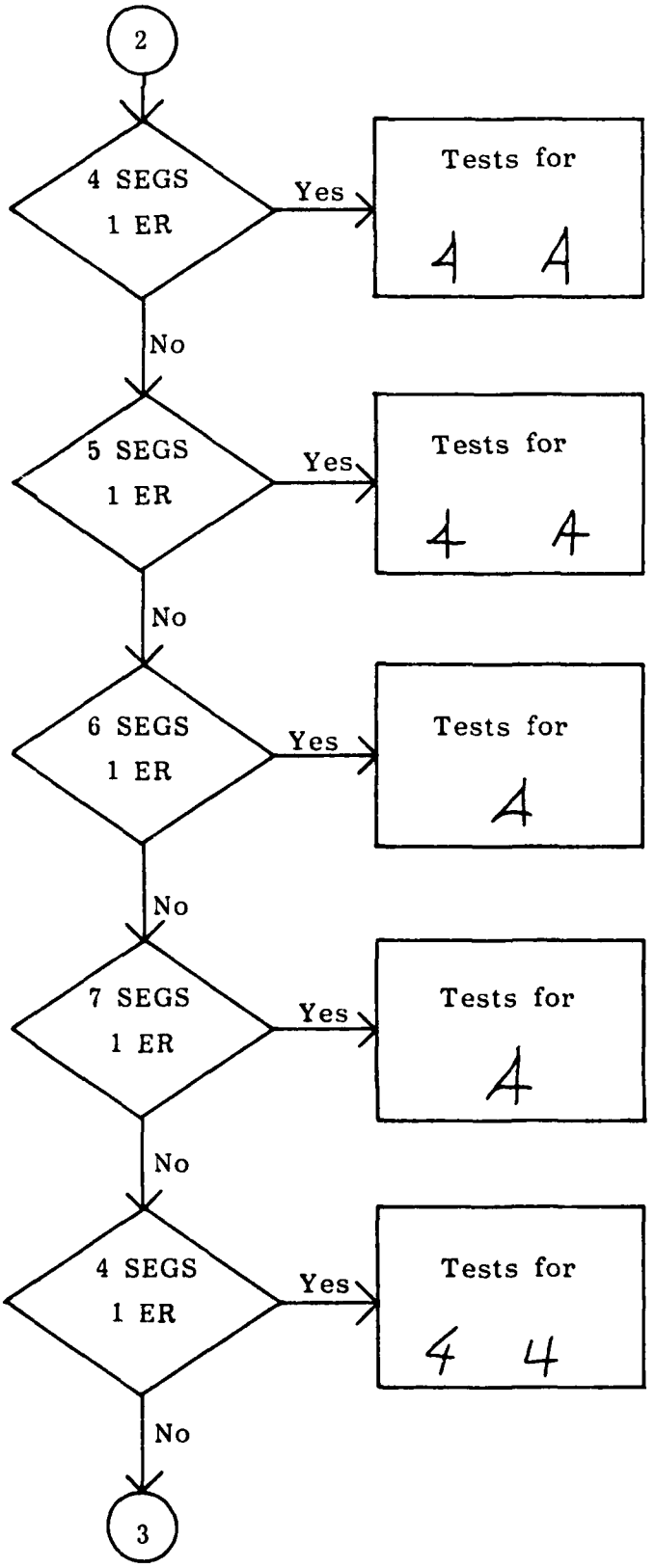


Figure 32, continued

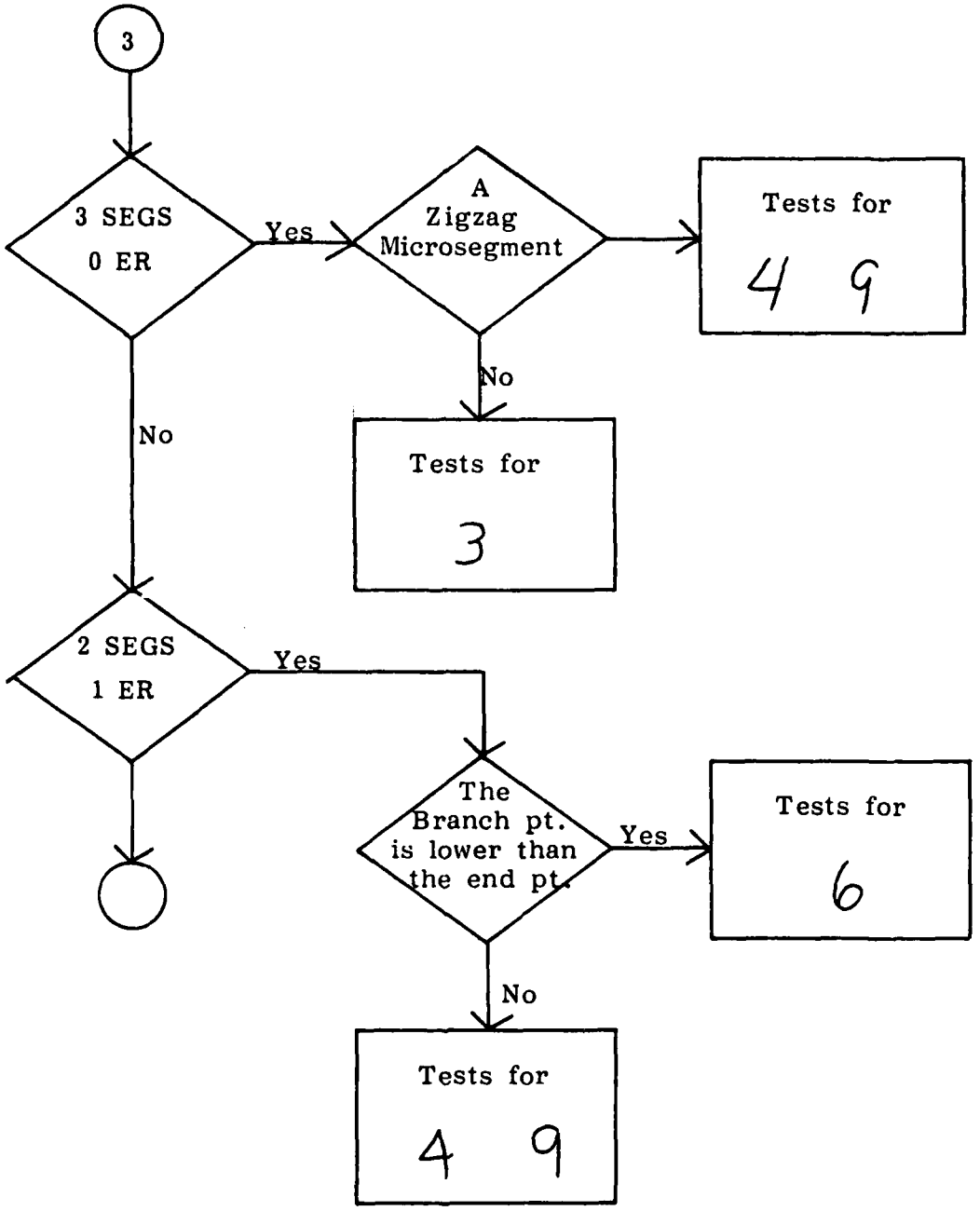


Figure 32, continued

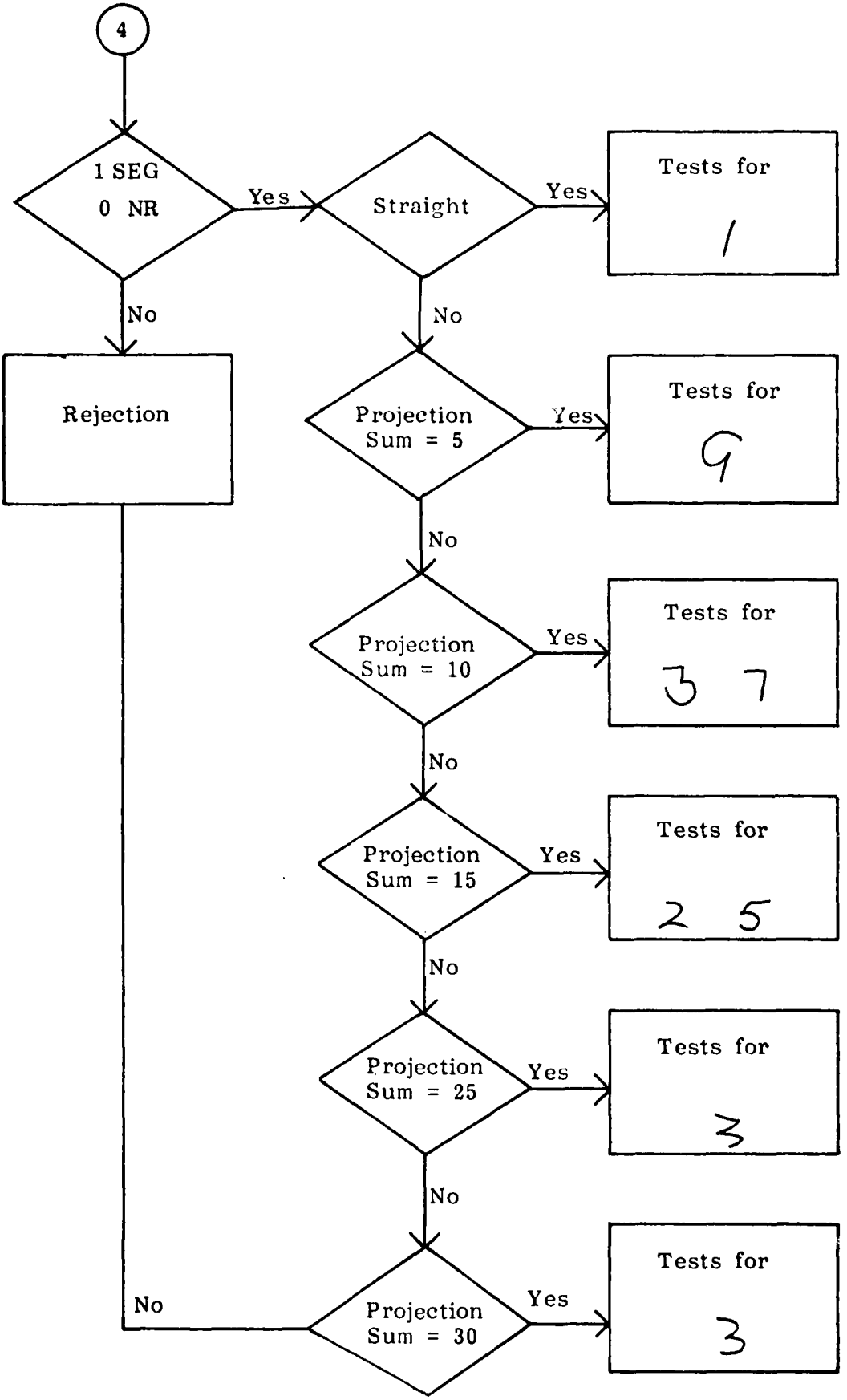


Figure 32, continued

pattern examiner classifies the input character as either unrecognizable or as its targeted numeral. These examiners use the specific shape measurements generated by the feature extraction modules. In general terms, they implement the heuristic handprint model for each character. The PAL TREE interpreter uses its "yes-no" pointer structure for determining the presence or absence of these targeted pattern properties.

5.0 CONCLUSIONS AND RECOMMENDATIONS

The advanced development efforts carried out by the NORDA Pattern Analysis Laboratory under the "Optical Character Recognition Algorithm Development" Subtask has provided DMA with software to "digitize" unconstrained, handprinted numerals appearing on a wide variety of DMA analog (graphic) documents. Furthermore, it has developed the new and necessary concepts and techniques for the implementation of software for other free-form handprint "digitizing" problems; e.g., geonames manuscripts. Finally, it has established a foundation for DMA, at the 6.3 funding level, for symbol recognition for other charting products that involve a free-form or unconstrained "layout." These techniques and algorithms, generally called Handprinted Symbol Recognition software, are a critical link in the automated cartography transformation process required for converting both archival and new source analog documents to an all-digital computer map and chart environment.

The HSR System, Version 2.0, has undergone development testing on the limited PAL database. These experiments have allowed checkout of the fundamental concepts, algorithms, and software modules for unconstrained, handprinted OCR. It has also indicated the need to perform extensive testing on DMA scanned documents. Current plans (FY-83) call for the generation of a significantly enlarged database of over 100,000 characters. The experiments using this data set will complete the testing for HSR, Version 2.0, for unconstrained numerals.

This system is also scheduled for experiments at DMA Centers during FY83-84 in conjunction with prototype character acquisition and document handling software to "digitize" smooth sheets and other similar graphic products, e.g., DFAD reference manuscripts. This interaction with DMA Center personnel should provide important feedback both (1) to the OCR development team concerning production center requirements and constraints and (2) to center personnel concerning new capabilities which are becoming available through OCR technology.

The software modules implementing the HSR algorithms have been programmed in standard FORTRAN. Therefore, they can be transported rather easily to other computer installations. In particular, the experimental testing to be performed at DMA Centers will utilize and exercise this feature. The source programs developed to implement the OCR algorithms are, of course, owned by the government and therefore can be modified or used for automated cartography applications on other MC&G tasks. Although this development software has not been optimized for speed, such "code streamlining" should be straight forward because of the modular development approach in a standard high-level programming language.

During the completion of the basic OCR techniques and concepts which lead to HSR, Version 2.0, an attack on several additional OCR problems has been initiated:

- The PAL investigation of unthinnable regions and the use of image boundary shape features will play a key role in extending the OCR techniques for the digital information capture of other unconstrained characters and symbols on map and chart products.
- The fundamental investigations of character recognition invariant to style/shape, size, and orientation has developed an important model of unconstrained handprinting and symbolic line-drawn information measurement. Such a model, required for the complex, unconstrained symbol problem, has led to initial study of techniques for easily adding new free-form symbol sets and the associated algorithm "training" methodologies required.
- Work is in progress on the target pattern categories for the free-form, handprinted alphabet.
- An analysis of the connected and disconnected character problem is currently under way.
- Work has begun on the use of context information for unconstrained handprinting. In particular, the relationship between the "word axis" and the "character axis" has received attention.
- A new area which has been considered and which should be explored further is a hybrid approach to complex documents. Such a system could "digitize" both fixed-font-like symbols appropriate for "template matching" technologies and "free-form symbol layouts" which require more sophisticated algorithms. This concept would optimize throughput, accuracy, and efficiency by employing a hierarchical approach to OCR. At any level in such a scheme, the system would use the least complex symbol recognition technique as possible as measured by a quality assurance module which would invoke a more complex algorithm when needed to meet the "digitizing" performance requirements. This approach can be based in part on the quality assurance and sufficient class membership concepts already incorporated in the HSR software.
- A final important issue already under investigation at the PAL is concerned with the proper partitioning of complex cartographic tasks between man and machine. In particular, it is clear that OCR technology must be utilized in a fail-safe manner in automated cartography, i.e., automated systems for "digitizing or reading" maps and chart documents must be "smart enough" to involve procedures that will flag the symbols which must be handled by humans through interactive editing. (This flagging capability has been built into HSR, Version 1.0 and 2.0.) Through this approach much of the time-consuming and labor intensive "digitizing" can be eliminated; at the same time, however, special cartographic judgment and expertise are utilized to resolve problems too complex for the computer recognition algorithms.

Completion of the development efforts in these areas will lead to software which can meet other DMA OCR requirements for

technology to convert graphic/image archival and source document data into computer compatible form for use in the DMA automated cartographic environment.

6.0 REFERENCES

Blesser, B., R. Shillman, C. Cox, T. Kuklinski, J. Ventura, and M. Eden (1973). Character Recognition Based on Phenomenological Attributes Visible Language, v. 7, n. 3, pp. 209-223.

Blesser, B. A., T. T. Kuklinski, and R. J. Shillman (1976). Empirical Tests for Feature Selection Based on a Psychological Theory of Character Recognition. Pattern Recognition, v. 8.

Blum, H. (1967). A Transformation for Extracting New Descriptions of Shape. Models for the Perception of Speech and Visual Forms, W. Wathen-Dunn, Ed.

Bolton, R. (1977). A Cartographic Optical Character Recognition System. University of Saskatchawan, Saskatoon, Saskatchawan.

Brown, R. M., M. Lybanon, and L.K. Gronmeyer (1979). Recognition of Handprinted Characters for Automated Cartography: A Progress Report. Proceedings of the SPIE, v. 205.

Brown, R. M. and L. K. Gronmeyer (1980). Recognition of Handprinted Characters for Automated Cartography. Technical Papers of the American Congress on Surveying and Mapping, 40th Annual Meeting, March.

Brown, R. M. (1981a). Handprinted Symbol Recognition System: A Key Element in Automated Cartography. National Ocean Survey Hydrographic Survey Conference. Norfolk, Va.

Brown, R. M. (1981b). Handprinted Symbol Recognition System: A Very High Performance Approach to Pattern Analysis of Free-Form Symbols. Proceedings IEEE Southeastern '81, April.

Brown, R. M. and C. F. Cheng (in prep.). Image Preprocessing for Handprinted Symbols for Automated Cartography. NORDA Technical Note 210.

Brown, R. M., C. L. Walker, and W. Osterman (1983). Raster Scan Character Recognition System. NORDA Technical Note 188.

Cheng, C. F. (1983). Image Preprocessing and Handwritten Character Recognition for Automated Cartography. Contract Report prepared for NORDA Code 371, by Computer Sciences Corporation, NSTL Station, Miss.

Dasarathy, B. V. and K. P. Bharath Kumar (1978). CHITRA: Cognitive Handprinted Input-Trained Recursively Analyzing System for Recognition of Alphanumeric Characters. Int. J. Computer and Information Sciences, v. 7.

DMA (1982a). Development of an Automated Cartographic Capability, The Final Report of the Automated Cartography Task Force.

Chairman, Dr. Jacob A. Teller, STT, Defense Mapping Agency Hydrographic/Topographic Center.

DMA (1982b). Automated Cartographic Capability for Hydrographic Products, the Final Report of the Hydrographic Cartography Study Group. Chairperson, Robert E. Sinclair, STT, Defense Mapping Agency Hydrographic/Topographic Center.

Duda, R. O. and Peter E. Hart (1973). Pattern Classification and Analysis. John Wiley and Sons, New York.

Gonzalez, R. C. (1980). Evaluation of the CHITRA Character Recognition System. Digital Decisions Systems, Inc., contract report prepared for the Pattern Analysis Laboratory, Naval Ocean Research and Development Activity.

Gonzalez, R. C. (1983). Syntactic/Semantic Techniques for Feature Description and Character Recognition. NORDA Technical Note 185.

Gronmeyer, L. K. and B. W. Ruffin (1978). An Application of Optical Character Recognition Techniques for the Digitization of Alphanumerics at the Defense Mapping Agency (DMA)--Part I. Proceedings, American Congress on Surveying and Mapping, 38th Annual Meeting.

Hilditch, C. J. (1969). Linear Skeletons from Square Cupboards. Machine Intell., v. 4, p. 403-420.

IPR (1978). DMA Memorandum for Record, dated 9 Jan 1979, Sub: In-Process Review (IPR) of Optical Character Recognition (OCR) Digitizer System Effort.

Lybanon, M. and L. K. Gronmeyer (1978). Recognition of Handprinted Characters for Automated Cartography. Proceedings of the SPIE, v. 155.

Lybanon, M. (1979). Preprocessing of CHITRA: Part I, Smoothing. Memorandum, Memo No. OCR-4-9-1, Computer Sciences Corporation, NSTL Station, Miss.

Martin, C. F. (1982). DMA Ltr. dated 28 Jan 1982, To: CAPT G. T. Phelps, USN, Commanding Officer, Naval Ocean Research and Development Activity, Dept. of the Navy, NSTL Station, Miss. Subj: Comments on 1981 Navy MC&G Technology Base R&D Review.

Macomber, M. M. (1977). DMA Ltr. dated 30 Nov 1977, To: Commanding Officer, Naval Ocean Research/Development Activity, Bay St. Louis, Miss. Subj: Redefinition of the Optical Character Reader (OCR) Effort.

Moritz, B. (1973). Inter-Office Memorandum, dated 21 Sept 1973, IRC Information Sciences Company. Subj: Algorithms for Live-Thinning.

Niesser, U. and P. Weene (1960). A Note on Human Recognition of Handprinted Characters. Information Control, 3:191-196, June.

OCR (1978). DMA Subtask Data Sheet (RDT&E), Optical Character Recognition Algorithm Development PE 63701B/3203/260, November.

OCR (1982). DMA Subtask Data Sheet (RDT&E), Optical Character Recognition Algorithm Development, PE 63701B/3203/260, March.

Overview (1978). An Overview of Optical Character Recognition (OCR) Technology and Techniques, June 7. Report prepared for the Defense Mapping Agency by the Naval Ocean Research and Development Activity and Computer Sciences Corporation, NSTL Station, Miss.

Pavlidis, T. (1980). Algorithms for Shape Analysis of Contours and Waveforms. IEEE Transactions on Pattern Analysis and Machine Intelligence, v. PAMI-2, n. 4, July.

RADC (1978). Statement of Work for Raster Scanned Character Recognition. PR No. I-8-4768, 27 Feb 78, Rome Air Development Center, Griffiss Air Force Base, New York.

RSCR (1982). DMA Subtask Data Sheet (RDT&E), Raster Scan Character Recognition, PE 63701B/3203/260, October.

Suen, C. Y., M. Berthod, and S. Mori (1980). Automated Recognition of Handprinted Characters--The State of the Art. Proceedings of the IEEE, v. 68, n. 4, April.

DISTRIBUTION LIST

Department of the Navy
Asst Secretary of the Navy
(Research Engineering &
System)
Washington DC 20350 (1)

Project Manager
ASW Systems Project (PM-4)
Department of the Navy
Washington DC 20360 (1)

Department of the Navy
Chief of Naval Material
Washington DC 20360 (1)

Department of the Navy
Chief of Naval Operations
ATTN: OP 951
Washington DC 20350 (1)

Department of the Navy
Chief of Naval Operations
ATTN: OP 952
Washington DC 20350 (1)

Department of the Navy
Chief of Naval Operations
ATTN: OP 980
Washington DC 20350 (1)

Director
Defense Technology Info
Cen
Cameron Station
Alexandria, VA 22314 (12)

Department of the Navy
Director of Navy
Laboratories
Rm 1062 Crystal Plaza
Bldg 5
Washington DC 20360 (1)

Commander
DWTaylor Naval Ship
R & D Cen
Bethesda MD 20084 (1)

Commanding Officer
Fleet Numerical Ocean Cen
Monterey CA 93940 (1)

Commander
Naval Air Development Cen
Warminster PA 18974 (1)

Commander
Naval Air Systems Command
Headquarters
Washington DC 20361 (1)

Commanding Officer
Naval Coastal Systems Cen
Panama City FL 32407 (1)

Commander
Naval Electronic Sys Com
Headquarters
Washington DC 20360 (1)

Commanding Officer
Naval Environmental
Prediction
Research Facility
Monterey CA 93940 (1)

Commander
Naval Facilities Eng Com
Headquarters
200 Stovall St.
Alexandria VA 22332 (1)

Commanding Officer
Naval Oceanographic Office
NSTL Station
Bay St. Louis, MS 39522 (1)

Commander
Naval Oceanography Command
NSTL Station MS 39522
Bay St. Louis, MS 39522 (1)

Director, Liaison Office
Naval Ocean R&D Activity
800 N. Quincy Street
502 Ballston Tower #1
Arlington VA 22217 (1)

Commander
Naval Ocean Systems Center
San Diego CA 92152 (1)

Superintendent
Naval Postgraduate School
Monterey CA 93940 (1)

Commanding Officer
Naval Research Laboratory
Washington DC 20375 (1)

Commander
Naval Sea System Command
Headquarters
Washington DC 20362 (1)

Commander
Naval Surface Weapons Cen
Dahlgren VA 22448 (1)

Commanding Officer
Naval Underwater Systems
Cen
ATTN: New London Lab
Newport RI 02840 (1)

Department of the Navy
Office of Naval Research
ATTN: Code 102
800 N. Quincy St.
Arlington VA 22217 (1)

Officer in Charge
Office of Naval Research
Detachment, Boston
Barnes Building
495 Summer St.
Boston, MA 02210 (1)

Commanding Officer
ONR Branch Office LONDON
Box 39
FPO New York 09510 (1)

Commanding Officer
ONR Western Regional Ofs
1030 E. Green Street
Pasadena CA 91106 (1)

Director
Scripps Inst of Oceanography
Univ of Southern California
La Jolla CA 92093 (1)

Working Collection
Texas A & M University
Department of Oceanography
College Station, TX 77843 (1)

President
Woods Hole Oceanographic Inst
Woods Hole, MA 20543 (1)

Director
Defense Mapping Agency
Washington, DC 20305 (5)

Director
Defense Mapping Agency
Hydrographic/Topographic Cen
6500 Brooke Lane
Washington, DC 20315 (8)

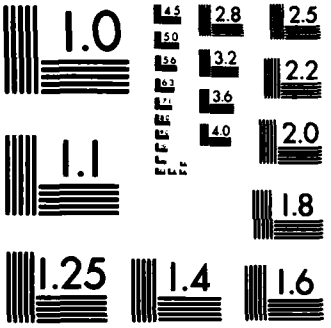
Director
Defense Mapping Agency
Aerospace Cen
St. Louis Air Force Station,
MO 63118 (5)

Director
Defense Mapping AGency
Special Program Office of
Exploitation and
Modernization
8301 Greensboro Drive,
Suite 1100
McLean, VA 22102 (2)

Commanding Officer
Naval Ocean R & D Activity
NSTL Station, MS 39529 (1)

AD-A127 566 OPTICAL CHARACTER RECOGNITION FOR AUTOMATED CARTOGRAPHY: THE ADVANCED DEV..(U) NAVAL OCEAN RESEARCH AND DEVELOPMENT ACTIVITY NSTL STATION MS.. 2/2

UNCLASSIFIED R M BROWN ET AL. MAR 83 NORDA-TN-187 F/G 8/2 NL



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NORDA Technical Note 187	2. GOVT ACCESSION NO. AD-A127566	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* Optical Character Recognition for Automated Cartography: The Advanced Development Handprinted Symbol Recognition System		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR*(s)* R.M. Brown C.F. Cheng		8. CONTRACT OR GRANT NUMBER*(s)*
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Ocean Research and Development Activity NSTL Station, Mississippi 39529		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 63701B
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Ocean Research and Development Activity NSTL Station, Mississippi 39529		12. REPORT DATE March 1983 13. NUMBER OF PAGES 93
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*		15. SECURITY CLASS. *(of this report)* UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Distribution Unlimited

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

This Naval Ocean Research and Development Activity (NORDA) Technical Note reviews the recent progress and present status of the Defense Mapping Agency (DMA) Subtask "Optical Character Recognition Algorithm Development" being carried out in the NORDA Pattern Analysis Laboratory (PAL), Mapping, Charting, and Geodesy (MC&G) Division. In particular, it describes the Handprinted Symbol Recognition System that is capable of reading and digitizing a wide range of isolated, unconstrained (free-form handprinted numerals appearing on various

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

(continued from Block 20)

DMA manuscript documents; e.g., smooth sheets, etc. Finally, the fundamental shape measurement and recognition tools incorporated in the HSR System can provide the foundation for other DMA systems to read the alphabet, foreign diacritics, and other map and chart symbols.

END

FILMED

5-83

DTIC